

**REPORT DOCUMENTATION PAGE**Form Approved  
OMB No. 074-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

<b>1. AGENCY USE ONLY (Leave blank)</b>		<b>2. REPORT DATE</b> June 14, 2007	<b>3. REPORT TYPE AND DATES COVERED</b> Final Performance Report Jun 2002 - Mar 2007	
<b>4. TITLE AND SUBTITLE</b> AFRL/Cornell Information Assurance Institute			<b>5. FUNDING NUMBERS</b> F49620-02-1-0170	
<b>6. AUTHOR(S)</b> Schneider, Fred				
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b>  Cornell University Ithaca, NY 14853			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b> 41467	
<b>9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b>  AFOSR Suite 325, Room 3112 875 Randolph Street Arlington, VA 22203-1768			<b>10. SPONSORING / MONITORING AGENCY REPORT NUMBER</b>  AFRL-SR-AR-TR-07-0432	
<b>11. SUPPLEMENTARY NOTES</b>				
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for Public Release; distribution is Unlimited				<b>12b. DISTRIBUTION CODE</b>
<b>13. ABSTRACT (Maximum 200 Words)</b> AFRL/Cornell Information Assurance Institute supports a broad spectrum of research and educational activities intended to enhance the science base for anticipated Air Force problems and to foster collaborations involving Cornell and AFRL/Rome researchers.				
<b>14. SUBJECT TERMS</b> Security, Trustworthiness, Assurance, Networks, Data Management				<b>15. NUMBER OF PAGES</b> 66
				<b>16. PRICE CODE</b>
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> Unlimited	

Final Report:  
AFRL/Cornell Information Assurance Institute

AFOSR Grant F49620-00-1-209

June 1 2002 – March 31 2007

Fred B. Schneider (Director)  
Department of Computer Science  
Cornell University  
Ithaca, New York 14853  
(607) 255-9221 (phone)  
(607) 255-4428 (fax)  
fbs@cs.cornell.edu

**Abstract**

The AFRL/Cornell Information Assurance Institute supported a broad spectrum of research aimed at developing a science and technology base to enhance information assurance and networked information systems trustworthiness—system and network security, reliability, and assurance. Moreover, the institute also fostered closer collaborations between Cornell and AFRL researchers, as well as facilitating technology transfer and exposing Cornell researchers to problems facing the Air Force.

## 1 Introduction

The AFRL/Cornell Information Assurance Institute (IAI) supported a broad spectrum of research and educational activities intended to enhance information assurance and networked information systems trustworthiness: system and network security, reliability, and assurance. IAI also fostered collaborations involving Cornell and AFRL researchers, with

- AFRL researchers able to participate in Cornell research projects, facilitating technology transfer as well as exposing Cornell researchers to problems facing the Air Force, and
- Cornell researchers able to become involved in AFRL projects and have access to unique AFRL facilities.

The research efforts encompassed two broad categories.

- (1) Research to develop a science base for anticipated Air Force problems. Much of this work was inspired by and had clear relevance to the Air Force Joint Battlespace Infosphere (JBI).
- (2) Research focused on problems identified by AFRL/Rome researchers. Each of these efforts typically involved collaboration with AFRL/Rome personnel.

A summary of the specific research undertaken appears below, in the next section; publications produced by IAI supported researchers appear after that. Figure 1 lists those researchers at Cornell (along with their specializations) whose work has been supported, in part, by IAI.

## 2 Summary of Research Accomplishments

### 2.1 Research to Expand the Science Base

**Ken Birman.** Birman and collaborators (Robbert van Renesse and Werner Vogels) pursued a three-pronged approach to developing a science base and technology for supporting large-scale reliable distributed systems. First, solutions to core problems were devised and shown correct. Next, those core solutions were embedded into widely used technology bases (such as the emerging Web Services architecture) so that practitioners and commercial users could see how the solutions fit into product lines and market visions. And finally, the solutions were tuned to offer performance competitive with

Kenneth Birman:	Distributed computing.
Richard Caruana:	Machine learning.
Robert L. Constable:	Applied logic and software assurance.
Alan Demers:	Database systems and database replication.
Paul Francis:	Networking.
Johannes Gehrke:	Database systems and data mining.
Joseph Y. Halpern:	Reasoning about knowledge and uncertainty.
Jon Kleinberg:	Algorithms and networks.
Dexter Kozen:	Program logics and semantics.
J. Gregory Morrisett:	Programming languages and security.
Andrew Myers:	Programming languages and security.
Robbert Van Renesse:	Distributed computing.
Radu Rugina:	Program analysis.
Fred. B. Schneider:	Trustworthy systems.
Emin Gun Sirer:	Distributed computing.
Jayavel Shanmugasundaram:	Database systems and data mining.

Figure 1: IAI Staff and Research Interests

or superior to that available in extant products. The mixture of performance and robustness increases the chances that the artifacts will win general acceptance, and the underlying theory and fundamental contributions of the work impact broad practice in the field.

Specifically, the group explored epidemic algorithms and peer-to-peer communication patterns to offer strong reliability properties and, at the same time, exhibit a degree of scalability missing from prior approaches. The vehicles for those investigations included the following.

- *Astrolabe* is a scalable monitoring and management framework for distributed systems settings.
- *Kelips* is a wide-area indexing scheme ( $O(1)$  lookups), that has been shown extremely robust against churn (a common form of denial of service attack).
- *Pbicast* is a new approach to reliable broadcast that offers probabilistic guarantees, has extremely low latency, is suitable for large-scale systems, and is quite robust.
- *QuickSilver* is a distributed computing platform for supporting high performance applications in which information replication is a central requirement (with or without additional requirements such as time-criticality, security, etc).
- *Ricochet* and *Plato*—respectively, a new time-critical multicast protocol and an ordering protocol—form the basis for the *Tempest* workbench that can be used to transform service oriented systems into clustered, scalable, robust versions.
- *Fireflies* is a scheme for building overlay networks in which nodes cooperate, peer-to-peer style, to ensure a high quality of video or other media delivery in a scalable manner despite internal and external attacks, including scenarios such as performance faults and outright Byzantine failures where nodes collaborate to disrupt the overlay.

This work is especially well suited for Air Force scenarios, such as those being considered for the Joint Battlespace Infosphere effort, where data is replicated because sensors need to report events to large numbers of clients that use the data to update tactical displays, formulate appropriate strategies, coordinate actions with coalition partners, or otherwise plan and act in reliable, trustworthy ways.

Experimental evaluation of these artifacts was undertaken using using a 250 node cluster (obtained under DURIP funding through AFOSR), PlanetLab, and other experimental resources.

**Rich Caruana.** Caruana works in the area of machine learning. Perhaps his most significant accomplishment under the auspices of IAI funding was a large-scale empirical comparison of ten learning methods: SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. The methods were evaluated on eight binary classification problems using nine different performance criteria: accuracy, squared error, cross-entropy, ROC Area, F-score, precision/recall break-even point, average precision, lift, and calibration. This is likely the largest empirical comparison of learning methods ever performed.

Some of the results are surprising. For example, while SVMs have excellent performance, their performance is not significantly better than other methods such as neural nets. Although neural nets and SVMs have excellent performance, tree ensemble methods such as boosted trees and random forests yield better results on metrics such as accuracy and AUC. However, when it comes to predicting probabilities, the best methods are neural nets and bagged decision trees. Interestingly, maximum-margin methods such as boosted trees and SVMs do not predict good probabilities even if their objective function is changed to log-loss.

No single learning method dominates the others. Each learning method, including simpler methods such as decision trees, naive bayes, and logistic regression, yields the best performance on at least one problem and evaluation metric. So it seems reasonable to conclude that no one (or even few) learning methods are yet able to do it all, and it still is important to consider the full portfolio of methods when the best performance is required.

Two methods for correcting the biased probabilities predicted by some learning methods—platt scaling and isotonic regression—were also examined. This involved measuring the kinds of distortions these calibration methods are most suitable for and also measuring how much data each method needs to be effective. The results suggest that after calibration with either of these methods, boosted trees, random forests, and SVMs predict the best probabilities, whereas neural nets and bagged trees predict the best probabilities prior to calibration. The results also show that Isotonic Regression requires more data than Platt Scaling to perform well, but yields better probabilities when more data is available. Platt Scaling outperforms

Isotonic Regression only when data is scarce.

This research also brought another surprise. As expected, different types of decision trees (e.g., CART, C4.5, ID3, Bayesian trees, Laplacian smoothed trees, ...) have significantly different performance on different problems and for different metrics. After bagging, however, the performance of all tree types is virtually identical, with the one exception that trees that prune heavily (e.g., CART trees) yield inferior performance after bagging because they introduce unwanted bias in order to improve variance, and bagging is able to reduce variance but not eliminate bias. Additional experiments will establish whether the same results apply for boosting.

Finally, a new unsupervised method for clustering, called Meta Clustering, was developed. Unlike supervised learning, unsupervised clustering is an ill-defined problem. In supervised learning, the availability of training targets allows one to define crisp performance criteria, such as accuracy and squared error. In clustering, these target labels are not available. This makes distinguishing between good and poor clusterings more difficult. Moreover, the utility of a clustering depends on how the clustering will be used, not on how well it predicts a pre-specified set of target labels.

Because it is difficult for users to examine even a single clustering, it is not feasible for users to examine many alternate clusterings to find the one that is most useful for their purposes. Instead, Meta Clustering clusters the diverse set of clusterings at a meta level so that similar clusterings are grouped together. The user is then presented with a very small set of qualitatively different clusterings from which to select, each representing the consensus of a meta clustering.

**Robert Constable.** Support for Constable's research led to further development and application of the Logic of Events. This is a very high level specification language for distributed computing tasks framed in computational type theory (CTT) and implemented in the Nuprl Logical Programming Environment. The distributed programs of this theory are called Message Automata (MA), and they are similar in some regards to the IO Automata of Lynch.

A collaboration with ATC-NY corporation produced a translator from Message Automata to Java, but a considerable part of the effort has been focused on security and, in particular, on pieces needed to verify the APSS asynchronous proactive secret sharing protocol. The APSS challenge problem led, for example, to including in computational type theory an account of randomness that has proven valuable both in specifying security proper-

ties and in consensus protocols as well.

In order to specify APSS in a natural way, a flexible distributed state machine capability had to be implemented. In advance of writing articles on topics such as randomness, secret sharing, and distributed state machines, these ideas were tested through formalization and application inside Nuprl. Experiments and small applications then led to improvements. The formal theory consists of over 3K objects (definitions, theorems, and specialized tactics). This is the equivalent of writing five completely formal books on the topic, each over six hundred pages long; it is posted on the Web.

Key concepts were also developed to make NuPRL practical for working on larger, more complex problems. These included:

- *Abstract Interfaces and Components.* DARPA's SAPIENT program is focused on intelligent adaptive middleware and, as part of this project, Robbert van Renesse designed a protocol for switching between different FIFO services. Rather than use out of band messages, the switching protocol makes use of the FIFO services it is switching between. Specification and verification of this switch led to the discovery of a very useful concept: an *abstract interface*. Using this concept, specifications of *components* can be written as constraints between input and output interfaces. And a general result has been proved that automates generation of code to forward data from one abstract interface to another, making code generation possible from a much higher level of abstraction than previously.
- *Logical Atoms, Security, and Randomized Algorithms.* *Logical atoms* can be used to model idealized versions of cryptographic systems. In order to have a theory of logical atoms, sound rules for the concept of independence had to be found—that is, rules for proving that an object  $x$  is independent of an atom  $a$ , which intuitively means that  $x$  does not contain atom  $a$  in any way. A by-product of developing this theory is a theory of independent random sequences, which in turn provides a formal basis for verifying algorithms that use randomization.

**Alan Demers.** Demers works on data replication, data stream processing and randomized gossip protocols. With Gehrke, Riedewald, and White he developed efficient query evaluation and multi-query optimization techniques for extensions of publish-subscribe systems. These extensions included stateful (parameterized) composite events and aggregation, both point-in-time and over temporal sequences. The techniques are based on

nondeterministic finite automata with data buffering and can be integrated with automata-based methods for (non-parameterized) XML filtering. The group also studied the formal underpinnings of such systems, developing an algebra of events that can describe parameterization, as well as temporal and spatial aggregation.

With Allavena and others, Demers investigated so-called gossip protocols for scalability and fault tolerance. In a *gossip protocol*, each node forwards messages to a small set of gossip partners chosen at random from the entire group membership. By discarding the strong reliability guarantees of traditional protocols in favor of probabilistic guarantees, gossip protocols can deliver greater scalability and fault tolerance. In early gossip algorithms, partners were chosen uniformly at random from the entire membership, limiting scalability because of the resources required to store and maintain complete membership views at each node. Later protocols avoided this issue by storing much smaller random subsets of the membership at each node, choosing gossip partners only from these local views. Such protocols are subtle: at least some local views must change in response to group membership changes in order to preserve connectivity and performance guarantees. While these protocols have been the subject of much simulation and analysis, Demers developed formal proofs of their properties—in particular, results for estimating the probability of partitioning.

**Paul Francis.** Francis is interested in computer networking and focuses on the network layer and overlay networking. He worked on Internet routing (BGP) scalability and stability, new architectures for DDoS and worm protection, and overlay approaches for providing IP anycast and multicast.

The lack of scalability and resulting questionable stability of the Internet's core routing protocol, BGP, has been a problem for years. Francis' approach to this problem is to minimize and focus the role played by BGP, making it easier to shift part of the routing problem from a hard task (a distributed route calculation) to a simple task (distributing a flat table). Specifically, in this new approach, BGP is used primarily to compute routes to major ISP POPs. Each POP is then responsible for knowing how to reach local destinations. Francis and graduate student Joy Zhang developed a simulation that tests the approach using an accurate map of the Tier-1 Internet topology. The simulation demonstrated that the size of BGP's computation could be reduced from nearly 200,000 entries to around 700. Further, it confirmed a reduction in the size of the total router forwarding table by an order of magnitude with less than 1% penalty in overall path

length.

Another issue Francis studied was DDoS attacks and worms. Here, he developed a solution in which special firewall-like boxes (called firebreaks), deployed more towards the core of the network, directly exchange IP packets with end hosts, and end hosts wishing to exchange packets must transmit the packets through these firebreaks. An IP addressing mode called IP anycast is used to route packets through a firebreak near the source host transparently to the source host. When a target host detects that it is under attack, the host instructs the appropriate firebreaks to take appropriate defensive measures, such as fair queuing based on end-host address. Likewise, when the firebreak system itself detects a spreading worm, it initiates appropriate defensive measures, such as dropping packets that match the worm signature.

IP anycast is a form of IP addressing and routing whereby multiple, geographically disperse endhosts share the same IP address. In collaboration with graduate student Hitesh Ballani, Francis designed and implemented an architecture, called PIAS (Proxy IP Anycast Service), for a simpler and more scalable IP anycast deployment. Through simulations, they showed that PIAS exhibits good stretch characteristics and scales well to group size and dynamics. In addition, Francis and Ballani did a measurement study of IP anycasted DNS root servers and showed that affinity (the ability for IP routing to select the same IP anycast destination over time) is quite good, reversing a decade-long held assumption to the contrary.

Francis also developed ChunkySpread, which uses random graph creation and simple localized tit-for-tat load-balancing rules to give end hosts fine-grained control over load while still allowing virtually limitless scalability by the number of participants. The result is an attractive alternative to End-System Multicast and to the more-traditional IP multicast protocols.

**Johannes Gehrke.** Gehrke is interested in data mining, and over the past two years his IAI-funded research has focused on three problem areas: constraints, privacy, and data stream change detection. The work in each of these areas has both a theoretical and practical components.

**Constraints.** Gehrke has produced algorithms that mine large datasets based on the subset of patterns that interest a user, and these algorithms exhibit orders of magnitude improvements in running time. The algorithms provide a unified view of data mining for constrained itemsets and, therefore, they provide a recipe for extending many extant algorithms to accommo-

date constraints.

**Privacy.** Gehrke introduced a new notion of privacy breeches and a methodology (called amplification) for limiting those breaches. Unlike previous work, privacy breaches can be bounded by using amplification—even in the absence of knowledge about the original data’s distribution.

A second area of investigation concerned data sharing, which can be broadly classified into internal data sharing and external data sharing. With internal data sharing, data is shared by users within an organization that controls access to the data; one major concern is ensuring data integrity as the data is updated. In external data sharing (also known as the data publishing), data is made publicly available for research or is shared with third parties for collaborative efforts; one major concern here is maintaining data privacy while at the same time permitting having the published data be accurate enough to detect trends or patterns in the data.

- For internal data sharing, Gehrke developed a method for verifying the integrity of a database that is delegated to a third-party. This database is shared between several users. As the third party may be susceptible to attacks, the query results it returns might be incorrect, and the data is susceptible to malicious modifications. Gehrke’s techniques use a relational representation of Merkle trees and communication of signatures at transaction commit time in order to prove to users that their query results are complete and authentic.
- For external data sharing, in recent years a new definition of privacy called  $k$ -anonymity has gained popularity. Gehrke showed with two simple attacks that a  $k$ -anonymized dataset has some subtle, but severe, privacy problems. This led Gehrke to propose a novel powerful privacy definition called  $\ell$ -diversity. In addition to building a formal foundation for  $\ell$ -diversity, experimental evaluations have demonstrated that  $\ell$ -diversity is practical and can be implemented efficiently.

**Change detection in data streams.** This problem is increasingly important as data from sensors is used in decision making applications. Gehrke developed techniques that provide guarantees on the statistical significance of detected changes. These methods also allow for a meaningful description and principled quantification of those changes. The methods are completely non-parametric; they require no prior assumptions on the nature of the distribution that generates the data (except that points in the

stream of generated independently). In addition, the methods work for both continuous and discrete data.

**Joseph Y. Halpern.** Halpern works on theoretical aspects of system security. Perhaps his most significant IAI-funded work has been the development of a logic (jointly with graduate student Victoria Weissman) and its embodiment as the Lithium language for reasoning about authorization policies. Such an authorization policy describes conditions under which an action is permitted or forbidden. Halpern and Weissman showed that a fragment of (multi-sorted) first-order logic can be used to represent and reason about policies. The use of first-order logic guarantees that policies have a clear syntax and semantics. Further restricting the fragment results in a language that is still quite expressive yet is also tractable. Questions about entailment, such as ‘May Alice access the file?’, can be answered in time that is a low-order polynomial (indeed, almost linear in some cases), as can questions about the consistency of policy sets. The Lithium language not only instantiates this theoretical policy work but it has been adopted by ARL for compliance checking on the MLWeb project.

Halpern also made progress (jointly with Ph.D. student Kevin O’Neill) on developing secrecy requirements for multiagent systems. Specifically, they developed a new, modal-logic based definition of secrecy that can handle probabilistic secrecy in a clean way and that suggests generalizations of secrecy that may be useful for dealing with resource-bounded reasoning and with issues such as downgrading of information.

Finally, Halpern made significant progress in understanding the nature of robust and resilient multiparty computation. In multiparty computation, each player  $i$  is given some private value  $x_i$  and wants to compute some function  $f(x_1, \dots, x_n)$  of these private values, without revealing the individual private values themselves. There are well-known protocols for doing this in the cryptography literature, but these protocols assume that no more than one third of the players are “bad” and that the remaining players follow the protocol. But Halpern, visitor Danny Dolev and his student Ittai Abraham, were able to show this computation can be performed without assuming players are “good” and “bad”—it suffices to assume that they have preferences such as to get the function value over not getting it or that others not get the value. With preferences, multiparty computation can be performed despite coalitions of size up to some fixed  $k$  with up to  $t$  players who behave unpredictably (perhaps irrationally), as long as  $2t + k < n$  holds, where  $n$  is the total number of players. This protocol requires cryptographic tech-

niques. A similar protocol exists that does not use cryptography, although it requires that  $3t + 2k < n$  hold. Moreover, these results have been proved optimal.

**Jon Kleinberg.** Kleinberg has been focusing on algorithms and models for complex networks, and on techniques for mining the information content of these networks. Two particular areas of interest are the temporal dynamics of information networks and the power of randomized probing algorithms to infer network properties.

- With Aizen, Huttenlocher, and Novak, Kleinberg used probabilistic models to track the popularity of downloadable items at the Internet Archive, which maintains a widely-used collection of freely available digital media. Performing the analysis at this scale required new algorithms for computing maximum-likelihood state sequences in hidden Markov models; these algorithms were presented in joint work with Felzenszwalb and Huttenlocher.
- With Sandler and Slivkins, Kleinberg developed new algorithms for detecting attacks on a network that leave it disconnected; the approach uses a small set of random probes and provides guarantees on the probability of misdetection. These results were extended in joint work with AFRL's Warren Debany to provide estimates, again with provable guarantees, on the full set of nodes rendered unreachable by such attacks. And in subsequent work with Slivkins and Wexler, Kleinberg considered an analogous problem for round-trip latencies in a network: using a small set of random probes, it is possible to reconstruct most pairwise latencies with small error. This provides the first step towards a theoretical explanation for the success of algorithms such as IDMaps (Francis *et al.*) and GNP (Ng and Zhang) that are used to infer latencies on the Internet.
- In work with Elliot Anshelevich, Anirban Dasgupta, Eva Tardos, Tom Wexler, and Tim Roughgarden, Kleinberg considered a model of self-interested agents who want to form a network connecting certain endpoints. The set of stable solutions—the Nash equilibria—may look quite different from the centrally enforced optimum. This work studied the quality of the best Nash equilibrium and refers to the ratio of its cost to the optimum network cost as the price of stability. The best Nash equilibrium solution has a natural meaning of stability in this

context—it is the optimal solution that can be proposed from which no user will defect.

In studying complex, large-scale social networks, it is important to analyze and accurately model the structure of the groups and communities embedded in them. The processes by which communities come together, attract new members, and develop over time is a central research issue in the social sciences—political movements, professional organizations, and religious denominations all provide examples of such communities. In the digital domain, on-line groups are becoming increasingly prominent due to the growth of community and social networking sites, such as MySpace and LiveJournal. However, the challenge of collecting and analyzing large-scale time-resolved data on social groups and communities has left most basic questions about the evolution of such groups largely unresolved.

In joint work with Lars Backstrom, Dan Huttenlocher, and Xiangyang Lan, Kleinberg investigated these questions using two large sources of data: friendship links and community membership on LiveJournal and co-authorship and conference publications in DBLP. They studied how the evolution of these communities relates to properties, such as the structure of the underlying social networks, and found the propensity of individuals to join communities and of communities to grow rapidly depends in subtle ways on the underlying network structure.

When monitoring spatial phenomena with wireless sensor networks, selecting the best sensor placements is a fundamental task. Not only should the sensors be informative, but they should also be able to communicate efficiently. In joint work with Andreas Krause, Carlos Guestrin, and Anupam Gupta, Kleinberg developed a data-driven approach that addresses three central aspects of this problem: measuring the predictive quality of a set of sensor locations (regardless of whether sensors are placed at these locations), predicting the communication cost involved with these placements, and designing an algorithm with provable quality guarantees that optimizes the tradeoff between quality and communication cost. Specifically, the approach uses data from a pilot deployment to build non-parametric probabilistic models called Gaussian Processes (GPs) both for the spatial phenomena of interest and for the spatial variability of link qualities, enabling the estimation of predictive power and communication cost of unsensed locations. The work, which received the Best Paper Award at the 2006 Conference on Information Processing in Sensor Networks (IPSN 2006), provides both provable theoretical guarantees as well as extensive experimental validation on several real-world placement problems, using a complete system imple-

mentation on 46 Tmote Sky motes, demonstrating significant advantages over existing methods.

**Dexter Kozen.** Kozen has been involved in the development of two tools for implementing high assurance software systems.

- The *BootSafe* system applies language-based techniques to ensure boot-time device drivers for plugin components do not contain malicious code.
- AESOP is a system for static analysis and runtime monitoring of untrusted software modules, such as commercial off-the-shelf (COTS) components, software from foreign sources, and web downloads. The main uses of AESOP are in combating malicious code (Trojan horses, spyware), enforcement of security policies, auditing of critical operations, behavior and performance profiling, and experimental application of heuristics for identifying malicious code.

In addition, Kozen pursued some more theoretical investigations involving Kleene algebra and its application to static analysis of programs. Kleene algebra with tests (KAT) is an algebraic system for program specification and verification that combines Kleene algebra, or the algebra of regular expressions, with Boolean algebra. It has been applied successfully in substantial verification tasks involving communication protocols, source-to-source program transformation, concurrency control, compiler optimization, and dataflow analysis.

With PhD student Kamal Aboul-Hosn, Kozen developed KAT-ML, an interactive theorem prover for KAT. The system is designed to support a natural style of equational reasoning. Source code and executable images for various platforms are publicly available. By focusing exclusively on KAT, it is possible to exploit the special features of the algebra to support a natural human-centered style of reasoning. For example, associativity is built in, and there are many other built-in heuristics and tactics particular to KAT. KAT-ML has been used to verify formally several known results in the literature, including the KAT translation of the Hoare partial correctness rules, a verification problem involving a Windows device driver, and an intricate scheme equivalence problem of Patterson that was previously only done by brute-force cut-and-paste on the flow graph. As the user applies proof rules, the system constructs an independently verifiable proof object, which would be useful in proof carrying code applications.

**Greg Morrisett.** Cyclone, developed by Morrisett and his graduate students, is a type-safe programming language based on C. It is a step towards the goal of providing language-based support to make legacy C code more secure. The type system of Cyclone accepts many C functions without change and uses the same data representations and calling conventions as C for a given type constructor. The Cyclone type system also rejects many C programs to ensure safety. For instance, it rejects programs that perform (potentially) unsafe casts, that use unions of incompatible types, that (might) fail to initialize a location before using it, that use certain forms of pointer arithmetic, or that attempt to do certain forms of memory management.

A semi-automatic tool was developed that can be used to automate the translation of legacy C programs to Cyclone. Programmers must touch about 10% of the code when porting from C to Cyclone, with most of the changes concerned with choosing pointer representations and only a very few changes that involve region annotations (around 0.6 % of the total changes).

In addition, to reduce Cyclone performance overheads associated with dynamic checks, a sophisticated intra-procedural analysis, which eliminates those checks, was also developed. A simple matrix-multiply now runs as fast as C code, where before it was taking over 5x as long.

Finally, new typing mechanisms were introduced to Cyclone in order to support a wider range of safe memory management options. Programmers had been restricted to using only garbage collection, stack allocation, or limited forms of region allocation, all of which could adversely affect time and space requirements, but support for dynamic region allocation, unique pointers, and reference-counted objects are now included. These mechanisms let programmers control memory management overheads without sacrificing safety. For instance, the throughput of the MediaNet streaming media server improved by up to 42% with decreased memory requirements from 8MB to a few kilobytes by virtue of these new features.

**Andrew C. Myers.** Myers' group explored the use of explicit information flow policies to build secure systems that enforce the confidentiality and integrity of information. This work is done in the context of the Jif programming language and compiler. Jif allows programmers to annotate programs with expressive security policies that are enforced at compile time.

There have been three areas of substantial progress. First, a new framework for building web applications in the Jif language has been prototyped. Second, prototype applications have been implemented using this frame-

work. Third, a theory of *decentralized robustness* has been developed as a way to describe how to use downgrading so that only intentional information release occurs.

**Secure servlet framework.** The group implemented a prototype framework for building web applications using the Jif programming language. In program source code, information is labeled with explicit annotations that define expressive, fine-grained policies for the confidentiality and integrity of that information, controlling to whom the information can be released, and who can affect the information. This security policy information for integrity and confidentiality can be computed at either compile time or run time, making it possible to make decisions (either in the program or under user control) based on the actual provenance of information.

With this software framework, called the Jif Servlet Framework (JSF), it is possible to implement a wide variety of web applications with much stronger assurance of confidentiality and integrity than is now possible. Explicit information security policies can be expressed either at compile time by the application implementer or deployer, or even at run time, possibly by the user. These information security policies can be visible to the user on the web browser so they can evaluate both whether the information they are viewing is trustworthy and whether the inputs they are entering will remain confidential.

By enforcing strong integrity policies and by controlling the HTML code produced by a web application, JSF protects against SQL injection attacks and cross-site scripting attacks. However, this is simply a side effect of enforcing more general integrity policies that are expressive enough to protect application users from each other, ensuring that a user's data is not modified by or leaked to users whom he does not trust.

Information flow is controlled by a combination of compile-time and run-time mechanisms. It is possible to inspect information at run time to determine whether it is too sensitive to be leaked and whether it has been tainted by sources that make it unsuitable for an intended use. Users of the system can tell by inspecting displayed web pages how the information they enter is used and how trustworthy the displayed information is. Because policies are attached to information as it propagates through the web application, information security in JSF is end-to-end. By contrast, secure operating systems control information flow only at coarse granularity and do not track information flows across a distributed system, as is necessary for web applications.

**Demonstration web applications.** Two interesting demonstration web applications have been developed using JSF: an email application specialized for cross-domain communication and a multi-user calendar system. These are the most sophisticated programs built to date using language-based information flow analysis, showing that the technique can be used even in programs that interact with a dynamically changing external environment and that add new security principals and policies at run time. Both applications allow users to define complex information security requirements, which are enforced by a combination of compile-time and run-time mechanisms.

**Jif language extensions.** To implement JSF, some significant new features were needed in the Jif programming language. The group has made these features available in the public release of a new major version of the language, Jif 3.0. Here, applications have the flexibility to implement their own notion of security principal and to define application-specific mechanisms for authorization and authentication. Jif 3.0 also improves Jif's ability to reason about dynamic security policies, letting applications express and enforce dynamic security requirements.

JSF and Jif 3.0 provide a framework in which secure web applications can be designed, implemented, and deployed. This framework is unique in giving assurance prior to deployment that web applications enforce strong, end-to-end confidentiality and integrity policies. The example applications demonstrate that this is a practical way to construct high-assurance web applications with complex security requirements.

**Robust information flow.** While Jif controls information flow and therefore gives strong assurance that information flow respects confidentiality and integrity policies, type systems, like that in Jif, are designed to enforce noninterference, a property that corresponds to zero information flow. However, real systems need to selectively release confidential information, and they need to selectively use untrustworthy information in trusted contexts. Jif provides an explicit *declassification* operator that can release information by relaxing confidentiality policies. As part of developing JSF, an explicit *endorse* operator has been added to Jif, allowing boosting of integrity policies. These two downgrading mechanisms are essential for building interesting systems in Jif.

Recent work by Chong and Myers subsequently showed that robustness can be practically implemented in the Jif language as part of the Jif type

system. Efficient static checking by the Jif type checker enforces the robustness condition. Moreover, robustness is enforced from the viewpoint of all users of the system, yielding a kind of decentralized robustness. This is important, because Jif is intended for building systems where users do not fully trust one another. Therefore, these users may disagree about what security policies need to be enforced and about who the potential attackers are. Not only is decentralized robustness now implemented in Jif, Myers' group has proved that a type system like the one implemented in Jif does enforce decentralized robustness.

**Radu Rugina.** Program analysis and compilation techniques were the subject of Rugina's investigations—specifically, the use of program analysis for memory optimizations and for the static detection of memory errors.

In connection with the Jreg project, Rugina's group developed a compilation and run-time system that provides region-based memory management for Java programs. Objects in this system are grouped together into regions, and all of the objects in each region are reclaimed at once, using explicit memory deallocation commands. Hence, the system doesn't require garbage collection and is well-suited for applications with real-time constraints.

The system includes two components:

- *Region compiler.* Given an input program, the compiler uses novel analyses and transformations to produce a semantically equivalent output program with region commands. The compiler guarantees memory safety and ensures that regions are deallocated as early as possible. Special region instructions have been added to the standard Java bytecodes, and the compilation system has been implemented as a bytecode-to-bytecode translator.
- *Region virtual machine.* The Kaffe open-source virtual machine was extended to support regions in two ways. First, the appropriate region run-time has been added and the VM interpreter has been extended to decode and execute region bytecode instructions. Second, a region verification algorithm has been developed and included in the bytecode verifier. Region bytecode verification ensures memory safety in the presence of (untrusted) explicit memory deallocation.

In another project, Rugina's group developed a novel approach to *shape analysis*, the analysis of heap-allocated linked structures. The analysis can successfully determine that standard destructive operations on recursive structures (such as insertions, deletions, rotations, or reversals) preserve the

tree shape or the acyclic list shape of these structures. The analysis has also been used in a tool aimed at finding memory errors in C programs. A few dozen memory leaks have been found in popular utility programs. The tool demonstrated that the analysis is efficient enough to scale to large programs, yet it is precise enough to yield a relatively low rate of false warnings.

**Jayavel Shanmugasundaram.** Shanmugasundaram's research focused on three areas: unifying the management of structured and unstructured data, peer-to-peer databases, and declarative web application development.

Along with graduate student Chavdar Botev and with Sihem Amer-Yahia of AT&T Research Labs, he developed a new query language called TeXQuery, which enables users to seamlessly query XML documents that contain a mix of structured and unstructured data. TeXQuery has had a surprisingly rapid and profound influence on the data management industry. The World Wide Web Consortium (W3C), the body that developed standards such as HTML and XML, and includes participation for all major data management vendors, adopted TeXQuery as the precursor to the standard query language for querying structured and unstructured XML data.

Shanmugasundaram was also involved in the development of the PEPPER peer-to-peer database system (joint with Gehrke). Current peer-to-peer (P2P) systems only support a very simple query interface over data; most are limited to simple equality lookups or keyword search queries. The main goal of PEPPER is to support sophisticated queries in a scalable and fault-tolerant P2P environment. Specifically, a new index structure called P-Ring has been developed; it can support range queries and XML queries. One of the interesting aspects of P-Ring is that it can provably guarantee that a user query will return correct results (under certain failure assumptions), even in a dynamic P2P environment where peers may continually fail, enter, or leave the system. P-Ring was implemented and tested on the PlanetLab cluster, which consists of more than 300 computers distributed all over the world.

Finally, Shanmugasundaram joint with Demers, Gehrke, and Ridewald developed Hilda, a high-level language for developing data-driven web applications. Developing data-driven web applications is a complex and challenging task, because the application development interface provided by existing platforms is often too low-level or does not provide a unified model for the whole application stack. Hilda addresses the above shortcomings by providing a high-level language for developing data-driven web applications. A prototype Hilda compiler, which translates high-level Hilda specifications

into executable code that runs on a standard 3-tier J2EE application server architecture, has also been developed.

**Fred B. Schneider.** Schneider does research both in fault-tolerance and security, with an eye to implementing systems that are trustworthy. The work typically has both theoretical and practical components.

Under the auspices of this grant (and jointly with Morrisett and Ph.D. student Kevin Hamlen), a new characterization was developed for what policies can be enforced using reference monitors and what policies can be enforced by static analysis and by program rewriting. In addition, a computability requirement on reference monitors was shown by the group to be necessary, but not sufficient, for obtaining a precise characterization of the class of policies actually realizable by reference monitors. Finally, these researchers identified a new property, called “benevolence” that provides an accurate upper bound on the power of reference monitors.

This theory of policy enforcement is embodied in an IRM (inlined reference monitor) rewriter for the Microsoft CIL that the group developed. It can perform arbitrary rewriting on that object code. Thus, it provides a way to retrofit security policies on programs that run on Microsoft operating systems.

Work in trying to understand how to combine fault-tolerance and security requirements, led not only to some interesting theory but also to two prototypes:

- CODEX (COrnell Data EXchange) is a distributed service for storage and dissemination of secrets. It was designed to be one of the components in a JBI-like secure publish/subscribe communications infrastructure, providing the support for storing secret keys used to encrypt published information objects and ensuring that (only) authorized subscribers retrieve those secret keys. Confidentiality, integrity, and availability of the secret keys being stored is crucial here; server failures must be tolerated; and attacks must be thwarted.
- CorSSO is a distributed multi-factor authentication service. It allows application servers to delegate client identity checking to combinations of authentication servers that reside in separate administrative domains. CorSSO authentication policies enable the system to tolerate expected classes of attacks and failures. A novel partitioning of the work associated with authentication of principals means that the system scales well with increases in the numbers of users and services.

Implementing trustworthy services is not simply a matter of adding security to a fault-tolerant system. Though server failures are often observed to be independent, attacks are a different story. An adversary that disrupts one replica will probably be able to exploit the same vulnerability at other replicas and disrupt them as well. So, although having  $2f + 1$  replicas might tolerate up to  $f$  faulty hosts, all hosts (hence all replicas) might succumb to a single attack. Eliminating software bugs eliminates vulnerabilities that would impinge on replica independence, but constructing bug-free software is quite difficult. This led Schneider to explore another means of increasing replica independence—diversity—and he concentrated on leveraging diversity that can be introduced automatically during compilation, loading, or in the run-time environment.

Progress here was made on two fronts. First, a characterization was obtained for the defensive power of mechanically-generated diversity. Specifically, a reduction was discovered from defenses created by mechanically-generated diversity to probabilistic dynamic type-checking. Thus, mechanically-generated diversity compares to a well-understood defense: strong typing. Second, revisiting replica-coordination protocols for the firewall prototype has led to new, practical algorithms.

**E. Gun Sirer** Sirer’s research was centered around the design and implementation of self-organizing systems. The work focused on settings where self-organization is critical—*ad hoc* networks and peer to peer overlays—and resulted in several prototypes: Magnetos, Meridian, Herbivore, Sextant, and Credence.

With MagnetOS, Sirer’s group built a new operating system for *ad hoc* and sensor networks that provides a simple, familiar and energy-efficient programming model to application developers for these emerging networks. MagnetOS provides a Java virtual machine abstraction over an entire network, transparently partitioning and moving application components for energy-efficient execution.

To experiment with MagnetOS, Sirer’s group developed a new simulation technique for high-performance, large-scale simulation of wireless networks. Called Staged Simulation, this technique is motivated by the observation that the limits to the scale and performance of current network simulators stem from redundant computations performed during simulation runs. Staged Simulation utilizes function caching (memoization) to record previous computations and reuse their results where appropriate.

The Meridian project for wide-area networks demonstrated how clients

could be routed to the nearest server for a networked service in a cheap, effective and scalable way. Meridian is a peer-to-peer overlay network for performing location-aware node and path selection in large-scale distributed systems. It builds a distributed, failure-resilient overlay graph, maintains the graph using a gossip protocol, and routes queries to suitable servers using a multi-hop search where the node at each hop exponentially reduces the distance to a target within the solution space. Three kinds of queries have been implemented using Meridian: (1) find the closest (lowest latency) node to a given target, (2) select the most centrally located node in a given set, and (3) discover a node within latency bounds of multiple targets.

Sirer’s group also examined how to hide the identity of communicating endpoints in wide-area networks and how to build stateless protocols that can be used to avoid denial-of-service attacks. The Herbivore project demonstrated how dining-cryptographer networks, which provide strong anonymity guarantees but are impractical, can be made scalable and deployed in practice. Herbivore can support audio-quality flows and hide the identity of the communicating endpoints even in a network where the adversary can eavesdrop on every link.

Sirer’s Sextant provided a comprehensive framework for determining the physical location of nodes and events in wireless networks. Determining the position of nodes without using GPS is a long-open problem, wherein previous solutions made only partial use of data available from the network layer. Sextant demonstrated how viewing the problem as a constraint satisfaction problem and aggressively extracting both positive (i.e. related to where nodes may be located) and negative (i.e. related to where nodes cannot be located) constraints from the network drastically reduces error. The Sextant system was deployed on sensor motes as well as laptops and PDAs, and it has been field-tested both in the Cornell arboretum and indoors in the lab.

Credence enables a peer to determine the authenticity of online content—that is, how accurate the purported description of the object matches the object itself. Participants in the Credence network vote on objects; Credence collates these votes, and weights them by a novel similarity measure, which weighs highly votes from like-minded peers while discounting the votes from peers engaged in vote-spamming. This novel voter-correlation scheme provides an incentive for peers to vote honestly and mitigates the impact of dishonest peers.

Credence is the first practical implementation of a peer-to-peer reputation scheme. Sirer’s group implemented Credence on top of the LimeWire client for the Gnutella network. This client provides a peer-based judgement

that a given object will possess the properties with which it is labeled and enables users to evaluate search results for authenticity before downloading. Unlike Google’s PageRank, the Credence trust metric is individualized to every participant and is therefore particularly resistant to attacks. In a 6-month study Sirer and his group conducted of the Credence trust graph, all but the smallest attacks (which were so small that they affected too few honest peers to be detected) were successfully identified.

## 2.2 Rome Labs Collaborations

**Declarative Distributed Sensor Databases.** This task was concerned with the investigation of a novel programming paradigm for sensor networks: Give users a declarative query interface to the sensor data, thereby abstracting the physical properties of the network when tasking the sensors. Such a *sensor database system* sends event data from source nodes to selected storage nodes, called *view nodes*, where the data is collected for further in-network processing. The research proceeded in two directions:

- Investigation of the problem of in-network compression.
- Development of a prototype system running on Berkeley motes with a generic architecture into which different compression algorithms can be inserted.

Results and code for data compression were delivered to a prototype system developed by Holzhauer’s group at Rome Labs; this prototype runs on the Pasta Nodes developed by ISI.

The project also investigated multi-query optimization for aggregate queries on sensor networks with development of a set of distributed algorithms for processing multiple queries that incur minimum communication while observing the computational limitations of the sensor nodes. The algorithms supported incremental changes to the set of active queries and allow for local repairs to routes in response to node failures. A thorough experimental analysis showed that this new approach results in significant energy savings, compared to previous work.

A novel scheduling protocol, Multi-Flow Power Scheduling (MPS), was also developed to coordinate transmissions of neighboring nodes such that interference and collision is minimized. MPS delivers data at a higher rate and consumes less energy as compared to existing approaches. However, MPS does not scale well to very large networks, due to the global nature of its scheduling. The scalability issue is addressed by combining MPS with

an existing algorithm from the literature FPS (Flexible Power Scheduling) to arrive at a hybrid solution that combines the best of both approaches. Hybrid Power Scheduling (HPS), achieves the scalability of FPS while still maintaining the higher data rate and energy efficiency of MPS. These protocols were implemented in TinyOS, and experiments showed that HPS and MPS have higher data delivery rates and are more energy efficient than other existing approaches.

**Pattern and Change Detection in Streaming Data.** The objective of this task was to develop new data mining models and algorithms for discovering structure in high-speed data streams. This includes abstract characterization of data types, methods of bit-stream segmentation, characterization of stream behaviors, and detection of possible phase transitions. The goal was a data mining system that never stops working, that incorporate new records immediately as they arrive, and that updates current and constructs new models continuously.

One area of focus was detection of a schema in a stream of records of unknown structure. A dynamic programming algorithm was developed that uses Minimum Description Length to decide between different segmentations in a principled way. The algorithm was implemented and tested on data from a synthetic data generator developed at Rome Labs.

Novel methods were also developed to construct summary data structures for online approximate query answers to estimate the sizes of spatial join queries across two data streams. In contrast to previous approaches, the techniques developed returned approximate results that come with provable probabilistic quality guarantees.

Ultimately, the entire task focus evolved into the design, implementation and evaluation of a stateful publish-subscribe system. The system produced processed a stream of events represented as XML information objects, and efficiently matched these against subscriptions that are specified using an extended version of the XQuery XML query language (the extensions allow users to specify subscriptions that depend on both the current event as well as previous events). Performance tests showed that the system could process an event over 100,000 structurally-similar stateful subscriptions in less than 50 milliseconds using a 933 MHz Pentium 3 Processor with 1GB of main memory.

The main technical challenge in developing a stateful publish-subscribe system was scaling up the number of subscriptions while still keeping event processing time of the order of a few milliseconds. The approach pursued

was to leverage existing relational database technology due to its impressive scalability and optimization capabilities. Specifically, techniques were developed for mapping XML information objects to relational tables so that state can be maintained efficiently as relations. Techniques were also developed for mapping stateful XML query subscriptions into relational triggers, thereby leveraging the sophisticated query optimization capabilities of relational database systems.

**Time-Sparse Pattern-based Intrusion Detection.** The goal of this task was to study the feasibility of building an Indications and Warning (I&W) system for cyber attacks; the challenge was to fuse and correlate the vast data store available from existing sensors in intrusion detection systems, firewalls, system audit logs, and network management systems. With machine learning and data mining methods, there was a chance that this challenge could be met.

The first phase of the project used retrospective analysis. Events in historical logs were correlated manually by human experts who labeled each event or combination of events as potential threat or not-threat. Machine learning was then used to find patterns in the data that are predictive of the threat. Part of the expert-labeled audit log data was held aside to use as test data to assess the performance of the system at recognizing potential threats. The data set included 41338 events from two weeks of network audit logs in 2003 during a period when significant potential threats to the network had been observed. The data appeared in two formats: single events and aggregations of events. Decision trees were selected to analyze the data. The decision trees were trained on subsamples of the data set and then evaluated on samples of the data held aside for testing. Care was taken not to allow events correlated with one potential attack to span both the training and test sets.

The performance of models trained using the best attributes ranged from 90% accuracy to 99% accuracy, depending on exactly which attributes were used and how the models were trained. This was significantly better than would be anticipated for this problem. But examination of the results suggested that the human experts who labeled the data used the same attributes available in the data as did the machine learning algorithms. Machine learning was thus able to learn rules similar to the rules the experts used to label the data. So if the expert-labeling is adequate, then it might be feasible to develop a pattern-recognition network intrusion I&W system; but if the expert labeling is incomplete, then an expert system trained on the current

data would not be effective.

The second phase of the project involved two intrusion detection data sets from Lincoln Labs. The first ID data set (LL1) focussed on detecting repeat attacks. The 2nd ID data set (LL2) focussed on novel attacks. Supervised machine learning was applied to LL1 using bagged probabilistic decision trees. Interesting results were obtained, as follows:

- probabilistic decision trees significantly outperformed traditional decision trees.
- bagging the probabilistic trees yielded substantial improvements in performance.
- a few individual trees generated for the bagged ensemble performed as well as the full ensemble, which has important implications for the ability to understand and verify the learned models.

A new clustering method developed for anomaly detection was applied to LL2 and these results were obtained:

- better performance at the low false positives end of the spectrum is obtained using methods that have poor performance for middle and high false positives, suggesting that it is critical to develop methods and metrics that are effective in the all-important low false positives regime.
- a feature representation reported in the literature to be effective on this problem does yield significant improvements in performance, but hurts performance if low false positives are needed.
- the ensemble clustering method we developed proved more effective at detecting novel kinds of attacks than we expected.

A new method was then developed for compressing complex, slow learned models into smaller, faster models that have comparable accuracy. The approach was to train the accurate but complex model first and then to use this model to provide pseudo-labels for a large synthetic data set, which is then used to train a new, much more compact, model. Experiments indicated that such “mimic models” could be more than 1000 times smaller and faster than the models they emulate, still with negligible loss in predictive accuracy. An additional order of magnitude improvement in execution time and memory footprint was achieved at a small loss in predictive accuracy.

**Survivable Publish/Subscribe Architectures.** This effort centered on developing proofs-of-concept for scalable publish-subscribe technologies aimed at the JBI system and other large-scale military information architectures. The team built a mock-up a Mercury-like system in which Astrolabe, Pbcast and Kelips were combined in support of the JBI publish-subscribe API, including all aspects of the emerging JBI security architecture. The prototype was used for detailed experimental studies and scenario evaluations needed to build confidence that scalability could be supported using this technology.

The task also explored the QoS space as it applied to publish-subscribe in GIG/JBI scenarios. Necessary QoS dimensions were identified, options for achieving such properties over existing publish-subscribe technologies were explored and architectures that could go beyond the limits of existing products were investigated.

**Characterizing the End-to-End Quality of Service Behavior of Networked Information Systems.** The objective of this task was to give a precise characterization for the end-to-end QoS behavior of networked systems and to verify critical QoS properties in a theorem proving environment.

A formal language was developed for specifying QoS properties of individual network components, the structure of a network and its effect on the QoS behavior of events, and requirements on the end-to-end QoS behavior of networked systems. The language is based on a logic of events, which in turn is embedded into the logical language of Cornell's Nuprl theorem proving environment. This formalizes the Lawrence QoS model and supports formal reasoning about the QoS of sets of executions with respect to the three basic measures: timeliness, accuracy, and precision. Example specifications of a QoS-aware router network were constructed using the language.

The collaboration also developed a strategy for formal reasoning about the QoS properties of (sets of events) between two arbitrary nodes in the network. The strategy determines the range of timeliness, accuracy, and precision that can be expected when events pass between these nodes. It can be used to determine whether a networked system satisfies a given set of the end-to-end QoS requirements. It can also be used to analyze the QoS behavior of specific network fragments.

To support compositional reasoning, formal *composition theorems* were stated and proved. These theorems uniformly describe the QoS effects of composing adjacent QoS-aware nodes or of adding a new node to a network with known QoS properties; proving the theorems (statically) with the Nuprl theorem proving environment reduces the computational burden

of (dynamically) analyzing the QoS behavior of a given network.

#### **Multi-query Optimization for Scalable Publish-Subscribe Systems.**

With JBI, a new data object (or event) has to be matched against a large set of existing subscriber queries or predicates. Optimizing and indexing predicates is far more complex than indexing data objects, but at the same time, the JBI needs to scale to a large number of queries.

To address this need, a novel event-workload aware data structure called the RPH-tree was developed in collaboration with AFRL researchers. The RPH-tree data structure is implemented in C++. Experiments demonstrated that it is capable of processing over 1000 events per second for over 500,000 subscription queries using a 2.7GHz Pentium processor with 1GB of main memory. For skewed workloads, RPH-trees outperform existing approaches by more than an order of magnitude. Three components were then transitioned into an AFRL codebase: (1) A synthetic subscription generator, (2) a synthetic event generator, and (3) a scalable adaptive publish/subscribe engine based on RPH-trees.

#### **Logic-based Security Policy Enforcement for Networked Services.**

This task investigated technologies for constructing secure web services. It introduced the WebGuard architecture whereby the security policy for a web service is specified separately from the service logic. A security enforcement tool parses the policy as well as the service implementation and automatically generates a secure service implementation with the requisite checks embedded on all code paths. A separate, abstract security policy specification enables enforcement to be performed in a systematic fashion through an inlined reference monitor, and this simplifies security enforcement by reducing the size of the trusted computing base.

A concrete implementation of the WebGuard architecture was completed for commonly-used web service platforms. This was achieved by implementing a parameterized platform specification, and it supports platform descriptions for .NET on Microsoft IIS, Tcl on AOLServer, and PHP on Apache. Performance experiments measured overheads within 1 to 2% of handcrafted, manually-secured web services.

**Encryption-based Authorization for Publish/Subscribe Architectures.** Military as well as enterprise applications increasingly are turning to networked infrastructures that allow publishers to disseminate confidential information to authorized subscribers. This task was concerned with de-

veloping cryptographic protocols to support publication services and their clients. Each publication service was assumed to have a public/private key pair, where the private key is shared and proactively refreshed periodically. Articles being stored on the publication service are encrypted using this public key. Clients encrypt articles and submit them to the publication service. The publication service, in turn, forwards these articles to its client and to other publication servers.

The development of a new distributed blinding protocol provided the basis for a distributed service to generate a blinding factor as well as two ciphertexts of the blinding factor—one for blinding and the other for unblinding. This is input for a re-encryption protocol in which data that has been encrypted under one key is transformed to being encrypted under another key but without ever appearing in an unencrypted form at any server. (Blinded data does appear in unencrypted form.)

A prototype distributed service was also built to store keys written by authorized distributors and make them available to authorized recipients, while preventing unauthorized principals from learning or altering these keys. This COrnell Data Exchange (CODEX) system employs the same basic “make only weak-assumptions” design philosophy that was implemented in COCA. No assumptions about message delivery-delays or execution speed are made, and compromised servers are presumed capable of arbitrarily malicious behavior.

**Firebreak IP DDoS Protection.** The only effective way to defend against distributed denial of service (DDoS) attacks is to filter out bad packets near the sources of the attack. But to implement this, many or most routers would have to be upgraded—an expensive (hence unrealistic) proposition. Therefore, this task explored an alternative, called Firebreak, that avoids the need for these upgrades by exploiting a mode of IP routing called IP anycast. Firebreaks work by forcing a level of indirection at the IP layer. Sources can reach a destination only by addressing packets to the Firebreak IP anycast address that represents the destination. This causes packets to be intercepted by firewall-like boxes (Firebreaks) near the source. The Firebreaks can then redirect the packets to the destination. If the destination determines that it is under attack, it can direct the appropriate Firebreaks to take defensive measures, thus protecting itself.

The three components of the firebreak system were designed and built: (1) the firebreak box deployed near the sources, (2) the DDoS detection monitor deployed at the destination, and (3) the anycast proxies needed

to deploy IP anycast. These components were deployed on an existing IP anycast deployment, and performance was measured. The experiments were conducted using 150 PlanetLab nodes, 250 traceroute servers, and over 30,000 DNS servers. Affinity (the ability for IP routing to select the same IP anycast destination over time) was found to be quite good, reversing a decade-long held assumption to the contrary. In addition, the experiments showed that IP routing selects poor paths in terms of latency.

**High-Performance, Attack-Resilient, Load-Balancing Distributed Hash Tables.** Distributed denial of service attacks on web sites are particularly hard to guard against, especially when mounted with large numbers of compromised hosts. Peer-to-peer, server-less systems make it possible to avoid this vulnerability in the first place. But the performance of existing peer-to-peer systems makes them unsuitable for critical, low-latency applications, and past attempts to use structured peer-to-peer systems for latency-sensitive applications, including DNS and the Web, have failed. Therefore, this task explored the feasibility of developing high-performance, adaptive, and scalable peer-to-peer systems.

The first system built, called Beehive, was designed to resist massive denial of service attacks. The system used proactive caching and replication to achieve  $O(1)$  lookup latency in peer-to-peer distributed hash tables. Using an analytical model that yields a closed form solution, it achieves a targeted level of performance with a minimum number of object replicas for ubiquitous Zipf-like query distributions. The Beehive approach enabled development of a replacement for legacy DNS, called CoDoNs.

The original Beehive approach was then extended by relaxing the assumptions that objects were all the same size, were not updated frequently, and requests followed a strict Zipf distribution. And the development of Honeycomb was undertaken. It is a self-organizing system that can serve as a foundation for other applications, such as web caches, publish-subscribe systems, distributed systems, and other similar services that operate by responding to queries. Honeycomb provides strong performance guarantees to applications, similar to Beehive, by optimally solving resource tradeoffs between bandwidth and performance. But whereas the Beehive approach relied on an analytical solution, Honeycomb uses a numerical solution to find the extent of replication and placement for each object. This technique enables Honeycomb to support objects of widely differing sizes, update rates, and non-Zipf query distributions.

Web traces were collected and the core result behind Honeycomb was

proved. A web cache called CobWeb based on the core Honeycomb installation was built and deployed. It was a success—CobWeb was receiving 45000 queries per day, and was used to shield popular websites (or data sources) from collapsing under load.

Finally, an effort was undertaken to measure the only widely deployed pub-sub system, RSS, and then to build a new system, based on an extension of Honeycomb, that supplants RSS and delivers updates to clients faster without additional load on servers. The measurement study spanned several months and involved more than a hundred users as well as more than 100,000 data channels. This study found that most aspects of a pub-sub system follow Zipf popularity distributions, indicating that Beehive-like techniques are well-suited to the problem domain. Further, it extracted parameters that could be used to build realistic pub-sub benchmarks.

On the system-building front, the Honeycomb framework was extended to perform intelligent resource allocation when nodes are monitoring a set of data sources. By judiciously deciding which data channels should be polled how often, Honeycomb is able to significantly reduce update latencies perceived by clients. A prototype of Honeycomb, called Corona, was deployed on Planetlab, and it quickly built up a user community of 300+ people who used it to monitor blogs and online news sources. An evaluation showed that Corona was able to reduce the average update latency from 30 minutes to 45 seconds, without incurring any additional load on the network. This unique application of self-organizing, peer-to-peer systems enables more efficient monitoring of online data sources.

**Minimally Invasive Network Monitoring.** This task addressed the problem of learning the topology of a network, or inferring its status, using minimally invasive methods. Two lines of investigation produced interesting results.

A new method was developed for topology discovery under a model of communication in which nodes broadcast to neighbors within range; the method can be initiated by an arbitrary “root” node in the network, and this method has the property that every node is the source of at most three broadcasts. Essentially, the method operates using an initial “outward” pass in which each node, upon first being contacted, performs a broadcast to awaken undiscovered neighbors; this is followed by an “inward” pass, in which a node waits to hear from all its newly-discovered neighbors before passing its own information back toward the root. Simulations showed the method significantly more communication-efficient than more naive flooding

schemes.

Second, work was undertaken to use random geometric graphs for reasoning about wireless networks. For many of the primary applications of wireless networks, the underlying environment has a large number of obstacles and communication can only take place among nodes when they are close in space and when they have line-of-sight access to one another. In such domains, the standard model of random geometric graphs is not a good approximation of the true constraints, since it is not designed to capture the line-of-sight restrictions. This led to a new random-graph model, incorporating both range limitations and line-of-sight constraints. Asymptotically tight results for  $k$ -connectivity were proved in this model. Further, this task showed that when the probability of node placement is a constant factor larger than the threshold for connectivity, then near-shortest paths between pairs of nodes can be found, with high probability, by an algorithm using only local information. In addition to analyzing connectivity and  $k$ -connectivity, the emergence of a giant component was studied, as was an approximation question, in which one seeks to connect a set of given nodes in such an environment by adding a small set of additional “relay” nodes.

**End-middle-end Internetworking.** The IP-centric end-to-end model of Internetworking has, in some ways, been an astonishing success. But the Internet and its requirements have grown to the point where the end-to-end IP address model of identifying end-points and applications is no longer adequate. To address the problems, this mini-task developed an end-middle-end approach to connection establishment that exploits another layer of indirection: the name. Of course names exist today, but they are not integrated into the network layer. The use of a rich name-based off-path signaling protocol allows hosts and middleboxes (i.e. firewalls) to negotiate policy, and the coupling of this off-path signaling with a light-weight in-path signaling protocol to establish connections.

The vision was implemented as a set of protocols, collectively called NUTSS. The system was deployed and was shown able to connect through firewalls with name-based policy rules, to negotiate the transport and encryption protocols, and to change IP addresses while maintaining a TCP connection. A culmination of the effort was establishment of an IRTF (Internet Research Task Force) research group to study the end-middle-end approach in a wider and more public context, with expectations that the effort will lead to standardization.

**Integrating Structured and Unstructured Data.** XQuery 1.0 and XPath 2.0 Full-text (XQFT) have been recently developed and proposed for extending XQuery and XPath with full-text search. While there have been many previous proposals to integrate XML queries and full-text search, an interesting aspect of XQFT is that it integrates full-text search with a full-function XML query language (XQuery). Consequently, it presents new challenges and opportunities for integrating sophisticated queries with fulltext search. The focus of this task was on development of a system called Quark, which addresses two such challenges: (1) performing keyword search over XML views, and (2) scoring arbitrarily complex combinations of XQuery and XQFT queries.

Documents in Quark are processed as followed. When an XML document is loaded into the system, it is processed by the Document Loader. The Document Loader first assigns unique Dewey IDs to the nodes of the document (a Dewey ID is a hierarchical numbering scheme where the ID of the node contains the ID of its parent node as a prefix). The document is then stored/indexed in three different formats: a compressed binary format in the Filesystem Storage, a Structure+Value Based Index (SVBI), and a Structure Based Inverted List Index (SBILI). Quark also supports keyword search over views, and it permits the integration of new scoring function for queries over structured and unstructured data. Version 1.0 of the system has now been released as open-source for deployment within the Air Force.

The other issue investigated was XML functional dependencies (XFDs) since they are prevalent in most XML data; they are important to areas such as key analysis, document normalization, and data integrity. XFDs are more complicated than relational functional dependencies because the set of XFDs satisfied by an XML document depends not only on the document values but also the tree structure and corresponding DTD. In particular, constraints imposed by DTDs may alter the implications from a base set of XFDs and may even be inconsistent with a set of XFDs. The interaction between XFDs and DTDs was examined and a sound and complete axiomatization for XFDs was investigated, both alone and in the presence of certain classes of DTDs. These DTD classes form an axiomatic hierarchy, with the axioms at each level being a proper superset of the previous. Furthermore, consistency checking with respect to a set of XFDs was shown feasible for these same classes.

### 3 Publications Supported under this Grant

- (1) Kamal Aboul-Hosn and Dexter Kozen. KAT-ML: An interactive theorem prover for Kleene algebra with tests. *Proceedings 4th International Workshop on the Implementation of Logics (WIL'03)*, Boris Konev and Renate Schmidt (eds.) (Almaty, Kazakhstan, September 2003), 2–12.
- (2) Kamal Aboul-Hosn and Dexter Kozen. An Interactive Theorem Prover for Kleene Algebra with Tests. *Journal of Applied Non-Classical Logics*, 16:1–2, 2006, 9–33.
- (3) Kamal Aboul-Hosn and Dexter Kozen. Relational Semantics for Higher-Order Programs. *Proceedings of the 8th International Conference on Mathematics of Program Construction (MPC'06)*, Kuressaare, Estonia, July 2006, 29–48.
- (4) Kamal Aboul-Hosn and Dexter Kozen. Local Variable Scoping and Kleene Algebra with Tests. *Proceedings of the 9th International Conference on Relational Methods in Computer Science and 4th International Workshop Applications of Kleene Algebra (RelMiCS/AKA'06)*, Manchester, UK, August 2006, 78–90.
- (5) I. Abraham, Y. Bartal, T-H. Chan, K. Dhamdhere, A. Gupta, J. Kleinberg, O. Neiman, and A. Slivkins. Metric Embeddings with Relaxed Guarantees. *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, 2005.
- (6) I. Abraham, D. Dolev, R. Gonen, and J. Y. Halpern. Distributed computing meets game theory: Robust mechanisms for rational secret sharing and multiparty computation. *Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing*, 2006, 53–62.
- (7) Frank Adelstein, Dexter Kozen, and Matt Stillerman. Malicious code detection for open firmware. *Proceedings 18th Computer Security Applications Conference (ACSAC'02)*, Dec. 2002, 403–412.
- (8) Atul Adya, Paramvir Bahl, Ranveer Chandra and Lili Qiu. Architecture and Techniques for Diagnosing Faults in IEEE 802.11 Infrastructure Networks. *ACM Mobiles 2004*, Philadelphia, PA.

- (9) Anastassia Ailamaki and J. E. Gehrke. Time management for new faculty. *SIGMOD Record*, Vol. 32, No. 2, (June 2003).
- (10) J. Aizen, D. Huttenlocher, J. Kleinberg, and A. Novak. Traffic-Based Feedback on the Web. *Proceedings of the National Academy of Sciences* 101 (Suppl.1):5254-5260, 2004.
- (11) A. Aliena, A. Demers and J. Hopcroft. Correctness of a Gossip Based Membership Protocol. *Proceedings of ACM Conference on Principles of Distributed Computing*, Las Vegas, NV, July 2005.
- (12) Sihem Amer-Yahia, Chavdar Botev, and Jayavel Shanmugasundaram. TeXQuery: A Full-Text Search Extension to XQuery. *World Wide Web Conference*, (May 2004), 583–594.
- (13) Rohit Ananthakrishna, Abhinandan Das, J. E. Gehrke, Flip Korn, Muthu Muthukrishnan, and Divesh Srivastava. Efficient approximation of correlated sums on data streams. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, No. 3 (May/June 2003), 569–572.
- (14) Elliot Anshelevich, Anirban Dasgupta, Jon M. Kleinberg, Eva Tardos, Tom Wexler, Tim Roughgarden. The Price of Stability for Network Design with Fair Cost Allocation. *45th Symposium on Foundations of Computer Science*, 2004, 295–304.
- (15) William Y. Arms, Selcuk Aya, Manuel Calimlim, Jim Cordes, Julia Deneva, Pavel Dmitriev, Johannes Gehrke, Lawrence Gibbons, Christopher D. Jones, Valentin Kuznetsov, Dave Lifka, Mirek Riedewald, Dan Riley, Anders Ryd, and Gregory J. Sharp. Three Case Studies of Large-Scale Data Flows. *Proceedings of the IEEE Workshop on Workflow and Data Flow for Scientific Applications* (SciFlow 2006), Atlanta, Georgia, April 2006, 66.
- (16) Benjamin Atkin and Kenneth P. Birman. Evaluation of an Adaptive Transport Protocol. *Proc. ACM INFOCOM 2003*, (San Francisco, April 1-3 2003).
- (17) Benjamin Atkin, Ken Birman. Network-Aware Adaptation Techniques for Mobile File Systems. *Proceedings of the 5th IEEE International Symposium on Network Computing and Applications* (IEEE NCA06), Cambridge, MA, June 2006.

- (18) Jay Ayres, J. E. Gehrke, Tomi Yiu, and Jason Flannick. Sequential pattern mining using bitmaps. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (Edmonton, Alberta, Canada), July 2002.
- (19) Lars Backstrom and Rich Caruana. C2FS: An Algorithm for Feature Selection in Cascade Neural Nets. *Proceedings of the IEEE World Congress on Computational Intelligence (IJCNN 2006)*, July 2006, 400–408.
- (20) L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group Formation in Large Social Networks: Membership, Growth, and Evolution. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.
- (21) Paramvir Bahl, Ranveer Chandra, and J. Dunagan. SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad-Hoc Wireless Networks. *ACM Mobiles 2004*, Philadelphia, PA.
- (22) P. Bahl, J. Y. Halpern, L. Li, Y.-M. Wang, R. Wattenhofer. A Cone-based Distributed Topology-control Algorithm for Wireless Multi-hop Networks. *IEEE/ACM Transactions on Networking*, **13**:1, 2005, 147–159.
- (23) Mahesh Balakrishnan and Ken Birman. Reliable Multicast for Time-Critical Systems. *Proceedings of the First IEEE Workshop on Applied Software Reliability (WASR 2006)*, Philadelphia, PA, June 2006.
- (24) Mahesh Balakrishnan, Stefan Pleisch, and Ken Birman. Slingshot: Time-Critical Multicast for Clustered Applications. *IEEE Network Computing and Applications 2005 (NCA 05)*, Boston, MA.
- (25) H. Ballani and P. Francis. Towards a Global IP Anycast Service. *Proceedings of ACM SIGCOMM* (Philadelphia, Pennsylvania, August 2005), 301–312.
- (26) Hitesh Ballani and Paul Francis. Towards a Deployable IP Anycast Service. *First Workshop on Real, Large Distributed Systems (Worlds '04)* (San Francisco, California, December 2004).
- (27) Rimom Barr, John C. Bicket, Daniel S. Dantas, Bowei Du, T.W. Danny Kim, Bing Zhou, and Emin Gün Sirer. On the need for system-level support for ad hoc and sensor networks. *Operating Systems Review* 36(2):1–5, ACM (April 2002).

- (28) Rimón Barr, Zygmunt Haas, and Robbert van Renesse. JiST: An Efficient Approach to Simulation using Virtual Machines. *Software—Practice and Experience*, 2004. Reprinted in *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks*, Jie Wu, editor, CRC Press, 2005.
- (29) Shai Ben-David, J. E. Gehrke, and Reba Schuller. A theoretical framework for learning from a pool of disparate data sources. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Edmonton, Alberta, Canada), July 2002.
- (30) K. Bhargavan, C. Fournet, A. D. Gordon, and R. Pucella. TulaFale: A security tool for web services. *Formal Methods for Components and Objects (FMCO 2003)*.
- (31) Mark Bickford and Robert L. Constable. A Casual Logic of Events in Formalized Computational Type Theory. In *Logical Aspects of Secure Computer Systems, Proceedings of International Summer School Marktoberdorf 2005*.
- (32) M. Bickford, R. L. Constable, J. Y. Halpern, and S. Petride. Knowledge-based Synthesis of Distributed Systems Using Event Structures. *Proceedings of the 11th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR 2004)*, 2005 (Lecture Notes in Computer Science, vol. 3452), 449–465.
- (33) David Biermann, Emin Gün Sirer, and Rajit Manohar. A Rate-Matching Approach to Dynamic Voltage Scaling. *Proceedings of First Watson Conference on the Interaction between Architecture, Circuits, and Compilers*, New York, October 2004.
- (34) Ken Birman. Can Web Services Scale Up? *IEEE Computer*, Vol.38, No.10, October 2005, 107–110.
- (35) Ken Birman. The Untrustworthy Services Revolution. *IEEE Computer* Vol.39 No.2, February 2006, 98–100.
- (36) Kenneth P. Birman. Like It or Not, Web Services are Distributed Objects! *Communications of the ACM*, Vol. 47, No. 12 (December 2004), 60–62.
- (37) Kenneth P. Birman. Bringing Autonomic, Self-Regenerative Technology into Large Data Centers. *New Developments in Software Develop-*

ment Workshop, (NDIST 04), St. John, US Virgin Islands, December 7-10, 2004.

- (38) Kenneth P. Birman. Reliable Distributed Systems Technologies, Web Services, and Applications. 2005, XXXVI, 668, Hardcover ISBN: 0-387-21509-3, 145.
- (39) Ken Birman. The League of SuperNets. *IEEE Internet Computing* 7, No. 5, 2003, 92–96.
- (40) Ken Birman, Coimbatore Chandrasekaran, Danny Dolev, and Robbert van Renesse. How the Hidden Hand Shapes the Market for Software Reliability. *Proceedings of the First IEEE Workshop on Applied Software Reliability*, Philadelphia, PA, June 2006.
- (41) Kenneth P. Birman, Jie Chen, Kenneth M. Hopkinson, Robert J. Thomas, James S. Thorp, Robbert van Renesse, and Werner Vogels. Overcoming Communications Challenges in Software for Monitoring and Controlling Power Systems. *Proceedings of the IEEE*, Vol. 9, No. 5, May 2005.
- (42) K. Birman, A. Demers, I. Gupta, D. Kostoulas, and D. Psaltoulis. Decentralized Schemes for Size Estimation in Large and Dynamic Groups. *Proceedings of the 4th IEEE International Symposium on Network Computing and Applications*, Cambridge, MA, July 2005.
- (43) Kenneth P. Birman, Saikat Guha, and Rohan Murty. Scalable, Self-Organizing Technology for Sensor Networks. *Advances in Pervasive Computing and Networking*, Bulent Yeler, ed., Springer Science+Business Media, Inc., New York, NY, 1–15.
- (44) Kenneth Birman, Robert Hillman, and Stefan Pleisch. Building Network-Centric Military Applications Over Service Oriented Architectures. *SPIE Defense and Security Symposium 2005*, (March 29-31, 2005), Orlando, FL.
- (45) Ken Birman, Robbert van Renesse, Werner Vogels. Navigating in the Storm: Using Astrolabe for Distributed Self-Configuration, Monitoring and Adaptation. *5th Annual International Active Middleware Workshop (AMS 2003)*, (Seattle, WA, June 2003).
- (46) Ken Birman, Robbert van Renesse, and Werner Vogels. Adding High Availability and Autonomic Behavior to Web Services. *Proceedings*

of the 26th Annual International Conference on Software Engineering (ICSE 2004) (Edinburgh, Scotland, May 2004).

- (47) Ken Birman, Robbert van Renesse, and Werner Vogels. Navigating in the Storm: Using Astrolabe to Adaptively Configure Web Services and Their Clients. *Cluster Computing Special Issue: Autonomic Computing* Vol.9, No.2, April 2006, 127–139.
- (48) L. Blume, D. Easley, and J. Y. Halpern. Redoing the foundations of decision theory. *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR 2006)*, 2006, 14–24.
- (49) Chavdar Botev, Sihem Amer-Yahia, and Jayavel Shanmugasundaram. A TeXQuery Based XML Full-Text Search Engine. *ACM SIGMOD Conference on Management of Data* (June 2004), 943–944.
- (50) Chavdar Botev, Hubert Chao, Theodore Chao, Yim Cheng, Raymond Doyle, Sergey Grankin, Jon Guarino, Saikat Guha, Pei Chen Lee, Dan Perry, Christopher Re, Ilya Rifkin, Tingyan Yuan, Dora Abdullah, Kathy Carpenter, David Gries, Dexter Kozen, Andrew Myers, David Schwartz, and Jayavel Shanmugasundaram. Supporting Workflow in a Course Management System. *Proceedings 36th Technical Symposium on Computer Science Education (SIGCSE '05)*, St. Louis, Missouri, Feb. 2005, ACM SIGCSE, 262–266.
- (51) Chavdar Botev and Jayavel Shanmugasundaram. Context-Sensitive Keyword Search and Ranking for XML. *Workshop on the Web and Data Bases*, Baltimore, VA, June 2005, 115–120.
- (52) Adrian Bozdog, Robbert van Renesse, and Dan Dumitriu. SelectCast—A Scalable and Self-Repairing Multicast Overlay Routing Facility. *Proceedings of the First ACM Workshop on Survivable and Self-Regenerative Systems*. (October 2003, Fairfax, VA.)
- (53) P.S. Bradley, J. E. Gehrke, R. Ramakrishnan, and R. Srikant. Philosophies and advances in scaling mining algorithms to large databases. *Communications of the ACM*, August 2002.
- (54) Cristian Bucila, Rich Caruana, and Alexandru Niculescu-Mizil. Model Compression. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2006)*, August 2006, 535–541.

- (55) Cristian Bucila, J.E. Gehrke, Daniel Kifer, and Walker White. DualMiner: A dual-pruning algorithm for itemsets with constraints. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Edmonton, Alberta, Canada), July 2002.
- (56) Cristian Bucila, J. E. Gehrke, Daniel Kifer, and Walker White. DualMiner: A dual-pruning algorithm for itemsets with constraints. *Data Mining and Knowledge Discovery*. Vol. 7, No 3 (July 2003), 241–272.
- (57) Doug Burdick, Manuel Calimlim, Jason Flannick, Johannes Gehrke, and Tomi Yiu. MAFIA: A Maximal Frequent Itemset Algorithm. *IEEE Transactions on Knowledge and Data Engineering*, Vol.17, No.11, November 2005, 1490–1504.
- (58) Doug Burdick, Manuel Calimlim, Jason Flannick, Johannes Gehrke, and Tomi Yiu. MAFIA: A Performance Study of Mining Maximal Frequent Itemsets. *Workshop on Frequent Itemset Mining Implementations (FIMI'03)* (Melbourne, Florida, November 2003).
- (59) M. Calimlim, J. Cordes, A. Demers, J. Deneva, J. Gehrke, D. Kifer, M. Riedewald and J. Shanmugasundaram. A Vision for PetaByte Data Management and Analysis Services for the Arecibo Telescope. *IEEE Data Engineering Bulletin*, Vol. 27, No. 4 (December 2004), 12–20.
- (60) Rich Caruana and Vidgina de Sa. Benefiting from the features that feature selection discards. *Journal of Machine Learning Research (JMLR) Special Issue on Feature Selection*, March 2003, 1245–1264.
- (61) Rich Caruana, Mohamed Elhawary, Daniel Fink, Wesley Hochachka, Steve Kelling, Art Munson, Mirek Riedewald, and Daria Sorokina. Mining Citizen Science Data to Predict Prevalence of Wild Bird Species. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD-2006), August 2006, 909–915.
- (62) Rich Caruana, Mohamed Elhawary, Nam Nguyen, and Casey Smith. Meta Clustering. *Proceedings of the Sixth International Conference on Data Mining* (ICDM'06), December 2006.
- (63) Rich Caruana, Thorsten Joachims, and Lars Backstrom. KDD-Cup 2004: Results and Analysis. *SIGKDD Explorations*, 6:2 (Fall 2004), 95–108.

- (64) Rich Caruana, Art Munson, and Alexandru Niculescu-Mizil. Getting the Most Out of Ensemble Selection. *Proceedings of the Sixth International Conference on Data Mining (ICDM'06)*, December 2006.
- (65) Rich Caruana and Alexandru Niculescu-Mizil. An Empirical Comparison of Supervised Learning Algorithms. *Proceedings of the 23rd International Conference on Machine Learning (ICML2006)*, June 2006, 161–168.
- (66) Ranveer Chandra, Paramvir Bahl and Pradeep Bahl. MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card. *Proceedings IEEE Infocom* (IEEE, Hong Kong, March 2004).
- (67) Ranveer Chandra, Kamal Jain, Mohammad Mahdian, and Lili Qiu. On the Placement of Internet Taps in Wireless Neighborhood Networks. *12th IEEE International Conference on Network Protocols, (ICNP 04)*, Berlin, Germany, October 5-8, 2004.
- (68) Ranveer Chandra and Daniel Mosse. Providing a bidirectional abstraction for unidirectional ad hoc networks. *INFOCOM 2002*, (New York City, New York), June 2002.
- (69) Moses Charikar, Jon M. Kleinberg, Ravi Kumar, Sridhar Rajagopalan, Amit Sahai, Andrew Tomkins. Minimizing Wirelength in Zero and Bounded Skew Clock Trees. *SIAM J. Discrete Math*, 17(2004), 582–595.
- (70) Zhiyuan Chen. *Building Compressed Database Systems*. Ph.D. Thesis, Cornell University, Department of Computer Science, August 2002.
- (71) James Cheney and Ralf Hinze. Poor man’s generics and dynamics. *Haskell Workshop 2002* (Pittsburgh, PA, September 2002).
- (72) James Cheney and Christian Urban. System description: Alpha-Prolog, a fresh approach to logic programming modulo alpha-equivalence. *Workshop on Unification* (Valencia, Spain, May, 2003).
- (73) Sigmund Chorem and Radu Rugina. A Verifier for Region-Annotated Java Bytecodes. *Proceedings 1st Workshop Bytecode Semantics, Verification, Analysis and Transformation (Bytecode '05)*, April 2005, Edinburgh, 167–184.

- (74) Sigmund Chorem and Radu Rugina. Region Analysis and Transformation for Java Programs. *Proceedings of the 4th ACM SIGPLAN International Symposium on Memory Management*, Vancouver, BC, Canada, October 2004, 85–96.
- (75) Hana Chockler and Joseph Y. Halpern. Responsibility and blame: A structural-model approach. *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI 2003)* (Acapulco, Mexico, 2003), 147–153.
- (76) Stephen Chong and Andrew C. Myers. Decentralized robustness. *Proceedings of the 19th IEEE Computer Security Foundations Workshop*, Venice, Italy, July 2006, 242–253.
- (77) Stephen Chong and Andrew C. Myers. Language-Based Information Erasure. *Proceedings 18th IEEE Computer Security Foundations Workshop*, June 2005, 241–254.
- (78) Stephen Chong and Andrew C. Myers. Security Policies for Downgrading. *Proceedings 11th ACM Conference on Computer and Communications Security*, October 2004, 198–209.
- (79) Francis C. Chu and Joseph Y. Halpern. Great expectations. Part I: On the customizability of generalized expected utility. *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI 2003)* (Acapulco, Mexico, 2003), 291–296.
- (80) Francis C. Chu and Joseph Y. Halpern. Great expectations. Part II: Generalized expected utility as a universal decision rule. *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI 2003)* (Acapulco, Mexico, 2003), 297–302.
- (81) Francis Chu, Joseph Halpern, and J. E. Gehrke. Least expected cost query optimization: What can we expect? *Proceedings of the 21st ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems* (PODS 2002), (Madison, Wisconsin), June 2002.
- (82) Michael R. Clarkson, Andrew C. Myers, and Fred B. Schneider. Belief in Information Flow. *Proceedings 18th IEEE Computer Security Foundations Workshop*, June 2005, 31–45.
- (83) Robert L. Constable. Information-Intensive Proof Technology. *Proof Technology and Computation*, editors H. Schwichtenberg and K. Spies, Kluwer, Amsterdam, 2005.

- (84) Robert L. Constable and Wojciech Moczydowski. Extracting Programs from Constructive HOL Proofs via IZF Set-Theoretic Semantics. *Proceedings of International Joint Conference on Automated Reasoning (IJCAR 2006)*, August 2006.
- (85) G.F.Cooper, V. Abraham, C. F. Aliferis, J M. Aronis, B. G. Buchanan, R. Caruana, M. J. Fine, J. E. Janosky, G. Livingston, T. Mitchell, S. Montik, and P. Spirtes. Predicting Dire Outcomes of Patients with Community Acquired Pneumonia. *Journal of Biomedical Informatics*, Vol.38, No.5, October 2005, 347–366.
- (86) Adina Crainiceanu, Prakash Linga, Johannes Gehrke, and Jayavel Shanmugasundaram. P-Tree: A P2P Index for Resource Discovery Applications. *Proceedings of the Thirteenth International World Wide Web Conference (New York, NY, May 2004)*.
- (87) Adina Crainiceanu, Prakash Linga, Johannes Gehrke, and Jayavel Shanmugasundaram. Querying Peer-to-Peer Networks Using P-Trees. *Seventh International Workshop on the Web and Databases (WebDB 2004)* (Paris, France, June 2004), 25–30.
- (88) Adina Crainiceanu, Prakash Linga, Ashwin Machanavajjhala, Johannes Gehrke, and Jayavel Shanmugasundaram. An Indexing Framework for P2P Systems. *World Wide Web Conference* (May 2004), 388–389.
- (89) Adina Crainiceanu, Prakash Linga, Ashwin Machanavajjhala, Johannes Gehrke, and Jayavel Shanmugasundaram. An Indexing Framework for Peer-to-Peer Systems. *ACM SIGMOD Conference on Management of Data* (June 2004), 939–940.
- (90) Karl Crary, Stephanie Weirich, and Greg Morrisett. Intensional polymorphism in type erasure semantics. *Journal of Functional Programming*, 12(6):567–600, November 2002.
- (91) Abhinandan Das, J. E. Gehrke, and Mirek Riedewald. Approximate join processing over data streams. *Proceedings of the 2003 ACM Sigmod International Conference on Management of Data (SIGMOD 2003)* (San Diego, California, June 2003).
- (92) Abhinandan Das, Johannes Gehrke, and Mirek Riedewald. Semantic Approximation of Data Stream Joins. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 1 (2005), 44–59.

- (93) Abhinandan Das, Indranil Gupta, and Ashish Motivala. SWIM: Scalable weakly-consistent infection-style process group membership protocol. *Proceedings of the International Conference on Dependable Systems and Networks 2002 (DSN 2002)*, June 2002, 303–312.
- (94) Abhinandan Das, Mirek Riedewald, and Johannes Gehrke. Approximation Techniques for Spatial Data. *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data (SIGMOD 2004)* (Paris, France, June 2004).
- (95) A. Dasgupta, J. Hopcroft, J. Kleinberg, M. Sandler. On Learning Mixtures of Heavy-Tailed Distributions. *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, 2005.
- (96) Alan J. Demers, Johannes Gehrke, Mingsheng Hong, and Mirek Riedewald. Processing High-Speed Intelligence Feeds in Real-Time. *IEEE International Conference on Intelligence and Security Informatics (ISI 2005)*, 617–618.
- (97) Alan Demers, Johannes Gehrke, Mingsheng Hong, Mirek Riedewald, and Walker White. Towards Expressive Publish/Subscribe Systems. *Proceedings of the 10th International Conference on Extending Database Technology* (EDBT 2006), Munich, Germany, March 2006, 627–644.
- (98) Alan Demers, Johannes Gehrke, Rajmohan Rajaraman, Niki Trigoni, and Yong Yao. The Cougar Project: A Work-In-Progress Report. *Sigmod Record*, 32 4, (December 2003).
- (99) Alan Demers, J. E. Gehrke, and Mirek Riedewald. Research issues in distributed mining and monitoring. *Proceedings of National Science Foundation Workshop on Next Generation Data Mining (NGDM 2002)*, (Baltimore, Maryland, November 2002).
- (100) Alan Demers, Johannes Gehrke, and Mirek Riedewald. The Architecture of the Cornell Knowledge Broker. *Proceedings of the Second Symposium on Intelligence and Security Informatics (ISI-2004)* (Tucson, Arizona, June 2004).
- (101) A. Demers, J. Gehrke, and M. Riedewald. Research Issues in Mining and Monitoring of Intelligence Data. *Data Mining: Next Generation Challenges and Future Directions*, K. Kargupta, A. Joshi, K. Sivakumar, and Y. Yesha eds., MIT/AAAI Press, 2005.

- (102) Alin Dobra, Minos Garofalakis, J. E. Gehrke, and Rajeev Rastogi. Processing complex aggregate queries over data streams. *Proceedings of the 2002 ACM Sigmod International Conference on Management of Data (SIGMOD 2002)*, (Madison, Wisconsin), June 2002.
- (103) Alin Dobra, Minos Garofalakis, Johannes Gehrke, and Rajeev Rastogi. Sketch-Based Multi-Query Processing over Data Streams. *Proceedings of the 9th International Conference on Extending Database Technology (EDBT 2004)*. (Heraklion-Crete, Greece, March 2004).
- (104) Alin Dobra and J. E. Gehrke. SECRET: A scalable linear regression tree algorithm. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Edmonton, Alberta, Canada), July 2002.
- (105) Alexandre Evfimievski, J. E. Gehrke, and Ramakrishnan Srikant. Limiting privacy breaches in privacy preserving data mining. *Proceedings of the 22nd ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS 2003)*, (San Diego, CA, June 2003).
- (106) Alexandre V. Evfimievski, Ramakrishnan Srikant, Rakesh Agrawal, and Johannes Gehrke. Privacy Preserving Mining of Association Rules. *Information Systems*, Vol. 29, No. 4 (2004), 343–364.
- (107) Alexandre Evfimievski, Ramakrishnan Srikant, Rakesh Agrawal, and Johannes Gehrke. Privacy preserving mining of association rules. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (Edmonton, Alberta, Canada), July 2002.
- (108) P. Felzenszwalb, D. Huttenlocher, and J. Kleinberg. Fast Algorithms for Large-State-Space HMMs with Applications to Web Usage Analysis. *Advances in Neural Information Processing Systems (NIPS) 16* (2003).
- (109) P. Francis. Is the Internet Going NUTSS? *IEEE Internet Computing* (Nov-Dec 2003), 96–99.
- (110) E. Friedman, J. Y. Halpern, and I. Kash. Efficiency and Nash Equilibria in a Scrip System for P2P networks. *Proceedings of the Seventh ACM Conference on Electronic Commerce*, 2006, 140–149.

- (111) John Funderburk, Gerald Kiernan, Jayavel Shanmugasundaram, Eugene Shekita, Catalina Wei. XTABLES: Bridging relational technology and XML. *IBM Research Journal* 41, No. 4 (Dec 2002), 616–641.
- (112) Wai Fu Fung, David Sun, and J. E. Gehrke. COUGAR: The network is the database. *Proceedings of the 2002 ACM Sigmod International Conference on Management of Data (SIGMOD 2002)*, (Madison, Wisconsin), June 2002.
- (113) Venkatesh Ganti, J. E. Gehrke, Raghu Ramakrishnan, and W.-Y. Loh. A framework for measuring changes in data characteristics. *Journal of Computer and System Sciences*, Vol. 64, No. 3, May 2002, 542–578.
- (114) Johannes Gehrke. Decision tree construction. *Handbook of Data Mining*, Nong Ye, Editor. Lawrence Erlbaum Associates, 2003.
- (115) Johannes Gehrke and Ling Liu. Guest Editors' Introduction: Sensor-Network Applications. *IEEE Internet Computing* 10(2), 2006, 16–17.
- (116) Nicholas Gerner, Fan Yang, Alan Demers, Johannes Gehrke, Mirek Riedewald, and Jayavel Shanmugasundaram. Automatic client-server partitioning of data-driven web applications. Description of system demonstration at SIGMOD 2006. *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data (SIGMOD 2006)*, Chicago, IL, June 2006, 760–762.
- (117) P. Ginsparg, J. Gehrke, and J. Kleinberg. Overview of the 2003 KDD Cup. *SIGKDD Explorations*, 2004.
- (118) Robert Givan, David McAllester, Carl Witty, and Dexter Kozen. Tarskian set constraints. *Information and Computation* 174, No. 2 (2002), 105–131.
- (119) Dan Grossman. Existential types for imperative languages. Type checking systems code. *Eleventh European Symposium on Programming* (Grenoble, France, April 2002), Lecture Notes in Computer Science Volume 2305, 21–35.
- (120) Daniel J. Grossman. *Safe Programming at the C Level of Abstraction*. Ph.D. Thesis, Cornell University, August 2003.
- (121) Daniel J. Grossman. Type-Safe Multithreading in Cyclone. *ACM Workshop on Types in Language Design and Implementation* (New Orleans, LA, January 2003).

- (122) D. Grossman, G. Morrisett, T. Jim, M. Hicks, J. Cheney, and Y. Wang. Region-based memory management in Cyclone. *ACM Conference on Programming Language Design and Implementation* (Berlin, Germany, June 2002), 282–293.
- (123) P. Grünwald and J. Y. Halpern. Updating probabilities. *Journal of AI Research* 19 (2003), 243–278.
- (124) P. Grünwald and J. Y. Halpern. When ignorance is bliss. *Proceedings of the Twentieth Conference on Uncertainty in AI* (2004), 226–234.
- (125) S. Guha and P. Francis. Characterization and Measurement of TCP Traversal through NATs and Firewalls. *Proceedings of Internet Measurement Conference (IMC)*, Berkeley, CA, October 2005, 199–211.
- (126) Saikat Guha, Rohan Murty, and Emin Gün Sirer. Sextant: A Unified Node and Event Localization Framework Using Non-Convex Constraints. *Proceedings of The International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*, Urbana-Champaign, Illinois, May 2005, 175–188.
- (127) S. Guha, T. Yutaka, and P. Francis. NUTSS: A SIP-based Approach to UDP and TCP Network Connectivity. *Proceedings ACM Future Directions in Network Architecture (FDNA) Workshop* (Portland, Oregon, August 2004).
- (128) Lin Guo, Jayavel Shanmugasundaram, Kevin Beyer, and Eugene Shekita. Efficient Inverted Lists and Query Algorithms for Structured Value Ranking in Update-Intensive Relational Databases. *Proceedings of the IEEE International Conference on Data Engineering*, Tokyo, Japan, April 2005, 298–309.
- (129) Lin Guo, Feng Shao, Chavdar Botev, Jayavel Shanmugasundaram. XRANK: Ranked keyword search over XML documents. *Proceedings of the ACM SIGMOD Conference on Management of Data* (Jun 2003), 16–27.
- (130) Indranil Gupta, Kenneth P. Birman, and Robbert van Renesse. Fighting fire with fire: Using randomized gossip to combat stochastic scalability limits. *Journal of Quality and Reliability Engineering International* (ed. Nong Ye), May/June 2002, Vol. 18, No. 3, 165–184.
- (131) Z. Haas, J. Y. Halpern, and L. Li. Gossip-based ad hoc routing. *Proceedings of Infocom*, 2002, 1707–1716.

- (132) Z. Haas, J. Y. Halpern, and L. Li. Gossip-based ad hoc routing. *IEEE/ACM Transactions on Networking* 14:3, 2006, 479–491.
- (133) Brian Hackett and Radu Rugina. Region-Based Shape Analysis with Tracked Locations. *Proceedings of the 32nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, Long Beach, CA, January 2005, 310–323.
- (134) M. M. Halldórsson, J. Y. Halpern, L. Li, and V. Mirrokni. On spectrum sharing games. *Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Distributed Computing* (2004), 107–114.
- (135) J. Y. Halpern. Sleeping Beauty reconsidered: Conditioning and reflection in asynchronous systems. *Oxford Studies in Epistemology*, Vol. 1 (T. S. Gendler and J. Hawthorne, eds.), 2005, 111–142.
- (136) J. Y. Halpern. *Reasoning About Uncertainty*, MIT Press, 2005. (Paperback edition of book originally published in 2003.)
- (137) J. Y. Halpern. Sleeping beauty reconsidered: Conditioning and reflection in asynchronous systems. *Ninth International Conference on Principles of Knowledge Representation and Reasoning (KR 2004)* (2004), 12–22.
- (138) J. Y. Halpern. Intransitivity and vagueness. *Ninth International Conference on Principles of Knowledge Representation and Reasoning (KR 2004)* (2004), 121–129.
- (139) J. Y. Halpern. From statistical knowledge bases to degrees of belief: An overview. *Proceedings of the 25th ACM Conference on Principles of Database Systems*, 2006, 110–113.
- (140) J. Y. Halpern and L. Li. A minimum-energy path-preserving topology-control algorithm. *IEEE Transactions on Wireless Communications* 3, 3 (2004), 910–921.
- (141) J. Y. Halpern and Y. Moses. Using counterfactuals in knowledge-based programming. *Distributed Computing* 17, 2 (2004), 91–106.
- (142) J. Y. Halpern and K. O’Neill. Secrecy in multi-agent systems. *Proceedings of the 15th IEEE Computer Security Foundations Workshop* (2002), 32–46.

- (143) Joseph Y. Halpern and Kevin O'Neill. Anonymity and information hiding in multiagent systems. *Proceedings of the 16th IEEE Computer Security Foundations Workshop* (Asilomar, CA, 2003), 75–88.
- (144) J. Y. Halpern and K. O'Neill. Anonymity and information hiding in multiagent systems. *Journal of Computer Security* 13:3, 2005, 483–514.
- (145) J. Y. Halpern and J. Pearl. Causes and explanations: A structural-model approach—Part I: Causes. *British Journal for the Philosophy of Science* 56:4, 2005, 843–887.
- (146) J. Y. Halpern and J. Pearl. Causes and explanations: A structural-model approach—Part II: Explanation. *British Journal for the Philosophy of Science* 56:4, 2005, 889–911.
- (147) Joseph Y. Halpern and Riccardo Pucella. Modeling adversaries in a logic for security protocol analysis. *Proceedings: Formal Aspects of Security*, (London, UK, 2002).
- (148) Joseph Y. Halpern and Riccardo Pucella. On the relationship between strand spaces and multi-agent systems, *ACM Transactions on Information and System Security* 6:1 (2003), 43–70.
- (149) Joseph Y. Halpern and Riccardo Pucella. Probabilistic algorithmic knowledge. *Proceedings of the Ninth Conference on Theoretical Aspects of Rationality and Knowledge* (Bloomington, Indiana, 2003), 118–130.
- (150) Joseph Y. Halpern and Riccardo Pucella. A Logic for Reasoning about evidence. *Proceedings of the Nineteenth Conference on Uncertainty in AI* 2003, 297–304.
- (151) J. Y. Halpern and R. Pucella. Probabilistic algorithmic knowledge. *Logical Methods in Computer Science* 1:3, 2005.
- (152) J. Y. Halpern and R. Pucella. Evidence With Uncertain Likelihoods. *Proceedings of the Twenty-First Conference on Uncertainty in AI*, 2005, 243–250.
- (153) J. Y. Halpern and R. Pucella. A logic for reasoning about evidence. *Journal of AI Research* Vol.26, 2006, 1–34.

- (154) J. Y. Halpern and L. C. Rêgo. Interactive Unawareness Revisited. *Proceedings of Tenth Conference on Theoretical Aspects of Rationality and Knowledge*, 2005, 78–91.
- (155) J. Y. Halpern and L. C. Rêgo. Extensive games with possibly unaware players. *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, 2006, 744–751.
- (156) J. Y. Halpern and L. C. Rêgo. Reasoning about knowledge of unawareness. *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR 2006)*, 2006, 6–13.
- (157) J. Y. Halpern and A. Ricciardi. A Knowledge-theoretic Analysis of Uniform Distributed Coordination and Failure Detectors. *Distributed Computing* **17**:3, 2005, 223–236.
- (158) J. Y. Halpern and R. Shore. Reasoning about common knowledge with infinitely many agents. *Information and Computation* **191**, 1 (2004), 1–40.
- (159) J. Y. Halpern and V. Teague. Rational secret sharing and multi-party computation. *Proceedings of 36th ACM Symposium on Theory of Computing* (2004), 623–632.
- (160) J. Y. Halpern and R. van der Meyden. A logical reconstruction of SPKI. *Journal of Computer Security* **11**, 4 (2003), 581–614.
- (161) J. Y. Halpern, R. van der Meyden, and M. Vardi. Complete axiomatizations for reasoning about knowledge and time. *SIAM Journal on Computing* **33**, 2 (2004), 674–703.
- (162) Joseph Y. Halpern and Victoria Weissman. Using first-order logic to reason about policies. *Proceedings of the 16th IEEE Computer Security Foundations Workshop* (Asilomar, CA, 2003), 187–201.
- (163) J. Y. Halpern and V. Weissman. A formal foundation for XrML. *Proceedings of the 17th IEEE Computer Security Foundations Workshop* (2004), 251–263.
- (164) K. Hamlen, G. Morrisett, and F.B. Schneider. Certified in-lined reference monitoring on .NET. *Proceedings of the ACM SIGPLAN Workshop on Programming Languages and Analysis for Security*, 2006, 7–16.

- (165) Kevin Hamlen, Greg Morrisett, and Fred B. Schneider. Computability classes for enforcement mechanisms. *ACM Transactions on Programming Languages and Systems* 28:1, January 2006, 175–205.
- (166) David Harel, Dexter Kozen, and Jerzy Tiuryn. Dynamic logic. In D. M. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic*, volume 4, Kluwer, 2nd edition, 2002, 99–217.
- (167) J. Hickey, A. Nogin, R. Constable, B. Aydemir, E. Barzilay, Y. Bryukhov, R. Eaton, A. Kopylov, C. Kreitz, V. Krupski, L. Lorigo, S. Schmitt, C. Witty, X. Yu. MetaPRL — A modular logical environment. *International Conference on Theorem Proving in Higher-Order Logics*, LNCS 2758, Springer-Verlag, 2003, 287–303.
- (168) K.M. Hopkinson, R. Giovanini, X. Wang, K.P. Birman, D.V. Coury, and J.S. Thorp. EPOCHS: Integrated COTS Software for Agent-based Electric Power and Communication Simulation. *2003 Winter Simulation Conference*, (December 2003, New Orleans).
- (169) Engin Ipek, Sally McKee, Bronis de Supinski, M. Schulz, and Rich Caruana. Efficiently Exploring Architectural Design Spaces via Predictive Modeling. *Proceedings of the 12th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, October 2006.
- (170) T. Jim, G. Morrisett, D. Grossman, M. Hicks, J. Cheney, and Y. Wang. Cyclone: A safe dialect of C. *Usenix Annual Technical Conference* (Monterey, CA, June 2002).
- (171) Dag Johansen, Håvard Johansen, Robbert van Renesse. Environment Mobility—Moving the Desktop Around. *Second International Workshop on Middleware for Pervasive and Ad-Hoc Computing*, (MPAC 2004), Toronto, Canada, October 2004.
- (172) Dag Johansen, Robbert van Renesse, and Fred B. Schneider. WAIF: Web of Asynchronous Information Filters. *Future Directions in Distributed Computing*, Lecture Notes in Computer Science, Volume 2585 (Schiper, Shvartsman, Weatherspoon, and Zhao, eds.) Springer-Verlag, 2003, 81–86.
- (173) William Josephson, Emin Gün Sirer, and Fred B. Schneider. Peer-to-peer authentication with a distributed single sign-on service. *Proceed-*

ings *Third International Workshop on Peer-to-Peer Systems (IPTPS'04)* (San Diego, CA, February 2004), ACM.

- (174) David Kempe, Alin Dobra, and J. E. Gehrke. Computing Aggregate Information using Gossip. *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2003)*, (Cambridge, MA, October 2003).
- (175) David Kempe, Jon M. Kleinberg, Alan J. Demers. Spatial gossip and resource location protocols. *J. ACM* 51(2004), 943–967.
- (176) David Kempe, Jon M. Kleinberg, Eva Tardos. Influential Nodes in a Diffusion Model for Social Networks. *Automata, Languages and Programming, 32nd International Colloquium*, 2005, 1127–1138.
- (177) Dan Kifer, Shai Ben-David, and Johannes Gehrke. Detecting Change in Data Streams. *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB 2004)* (Toronto, Canada. August 2004).
- (178) Daniel Kifer and Johannes Gehrke. Injecting Utility into Anonymized Datasets. *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data (SIGMOD 2006)*, Chicago, IL, June 2006, 217–228.
- (179) Daniel Kifer, J. E. Gehrke, Cristian Bucila, and Walker White. How to quickly find a witness. *Proceedings of the 22nd ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS 2003)*. (San Diego, CA, June 2003).
- (180) J. Kleinberg. Bursty and Hierarchical Structure in Streams. *Data Mining and Knowledge Discovery* 7, 4 (2003), 373–397.
- (181) J. Kleinberg. The Small-World Phenomenon and Decentralized Search. *SIAM News* 37, 3 (April 2004).
- (182) J. Kleinberg. Analyzing the scientific literature in its online context. *Nature Web Focus on Access to the Literature* (April 2004).
- (183) J. Kleinberg. An Approximation Algorithm for the Disjoint Paths Problem in Even-Degree Planar Graphs. *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, 2005.

- (184) J. Kleinberg. The World at Your Fingertips. *Nature (Books and Arts)* 440(2006), 279.
- (185) J. Kleinberg. Complex Networks and Decentralized Search Algorithms. *Proceedings of the International Congress of Mathematicians (ICM)*, 2006.
- (186) J. Kleinberg, C. Papadimitriou, and P. Raghavan. Segmentation problems. *Journal of the ACM* 51, 2 (2004) 263–280.
- (187) J. Kleinberg and P. Raghavan. Query Incentive Networks. *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, 2005.
- (188) J. Kleinberg and M. Sandler. Using Mixture Models for Collaborative Filtering. *Proc. 36th ACM Symposium on Theory of Computing* (2004).
- (189) J. Kleinberg, M. Sandler, and A. Slivkins. Network Failure Detection and Graph Connectivity. *Proceedings 15th ACM-SIAM Symposium on Discrete Algorithms* (2004), 76–85.
- (190) Jon M. Kleinberg, Aleksandrs Slivkins, and Tom Wexler. Triangulation and Embedding Using Small Sets of Beacons. *45th Symposium on Foundations of Computer Science*, 2004, 444–453.
- (191) Robert D. Kleinberg and Jon M. Kleinberg. Isomorphism and embedding problems for infinite limits of scale-free graphs. *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2005, 277–286.
- (192) D. Koller and J. Y. Halpern. Representation dependence in probabilistic inference. *Journal of AI Research* 21 (2004), 319–356.
- (193) Dionysios Kostoulas, Dimitrios Psaltoulis, Indranil Gupta, Ken Birman, and Al Demers. Decentralized Schemes for Size Estimation in Large and Dynamic Groups. *IEEE Network Computing and Applications 2005* (NCA 05), Boston, MA, October, 2005.
- (194) Lucja Kot and Dexter Kozen. Kleene Algebra and Bytecode Verification. *Proceedings 1st Workshop Bytecode Semantics, Verification, Analysis and Transformation (Bytecode '05)*, April 2005. Edinburgh: Fausto Spoto, 201–215.

- (195) Dexter Kozen. On the complexity of reasoning in Kleene algebra. *Information and Computation* 179 (2002), 152–162.
- (196) Dexter Kozen. On Hoare logic, Kleene algebra, and types. In P. Gärdenfors, J. Woleński, and K. Kijania-Placek, editors, *In the Scope of Logic, Methodology, and Philosophy of Science*, Volume 1 of the 11th International Congress Logic, Methodology and Philosophy of Science, (Cracow, August 1999), volume 315 of *Studies in Epistemology, Logic, Methodology, and Philosophy of Science*, Kluwer (2002), 119–133.
- (197) Dexter Kozen. Some results in dynamic model theory (abstract). In E. A. Boiten and B. Möller, editors, *Proceedings Conference on Mathematics of Program Construction (MPC'02)*, Lecture Notes in Computer Science Volume 2386 (July 2002), 21.
- (198) Dexter Kozen. Automata on guarded strings and applications. *Matemática Contemporânea* 24 (2003), 117–139.
- (199) Dexter Kozen. Automata on guarded strings and applications. *Matemática Contemporânea* 24, 2003, 117–139.
- (200) Dexter Kozen. Computational inductive definability. *Annals of Pure and Applied Logic* 126 (2004), 139–148.
- (201) Dexter Kozen, (ed.). *Proceedings 7th International Conference on Mathematics of Program Construction (MPC 2004)* (Stirling, Scotland, July 2004), vol. 3125 of *Lecture Notes in Computer Science*, Springer-Verlag, 400 pages.
- (202) Dexter Kozen. Some results in dynamic model theory. *Science of Computer Programming* 51 (2004), 3–22.
- (203) Dexter Kozen. *Theory of Computation*. Springer-Verlag, New York, 2006.
- (204) Dexter Kozen. Coinductive Proof Principles for Stochastic Processes. *Proceedings of the 21st Symposium on Logic In Computer Science (LICS'06)*, Seattle, Washington, August 2006, 359–366.
- (205) Dexter Kozen. On the Representation of Kleene Algebras with Tests. *Proceedings of the Conference on Mathematical Foundations of Computer Science (MFCS'06)*, August 2006, 73–83.

- (206) Dexter Kozen, Christoph Kreitz, and Eva Richter. Automating Proofs in Category Theory. *Proceedings of the 3rd International Joint Conference on Automated Reasoning (IJCAR'06)*, Seattle, Washington, August 2006, 392–407.
- (207) Dexter Kozen and Matt Stillerman. Eager class initialization for Java. In W. Damm and E.R. Olderog, editors, *Proceedings 7th International Symposium on Formal Techniques in Real-Time and Fault Tolerant Systems (FTRTFT'02)*, Lecture Notes in Computer Science, Volume 2469, Springer-Verlag, Sept. 2002, 71–80.
- (208) Dexter Kozen and Jerzy Tiuryn. Substructural logic and partial correctness. *Trans. Computational Logic* 4, No. 3 (2003), 355–378.
- (209) A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg. Near-optimal Sensor Placements: Maximizing Information while Minimizing Communication Cost. *Proceedings of Information Processing in Sensor Networks (IPSN)*, 2006.
- (210) Christoph Krietz. Designing reliable, high-performance networks with the nuprl logical programming environment. *2002 AAAI Spring Symposium on Logic-Based Program Synthesis* (March 2002).
- (211) C. Kreitz. Building reliable, high-performance networks with the NuPRL proof development system. *Journal of Functional Programming*, 2003.
- (212) Christoph Kreitz. Building Reliable, High-Performance Networks with the Nuprl Proof Development System. *Journal of Functional Programming*, Vol. 14, No. 1, 2004, 21–68.
- (213) Christoph Kreitz and Heiko Mantel. A Matrix Characterization for Multiplicative Exponential Linear Logic. *Journal of Automated Reasoning*, Vol. 32, No. 2, 2004, 121–166.
- (214) Ashwin Kumar, V. Machanavajjhala, and Johannes Gehrke. On the Efficiency of Checking Perfect Privacy. *Proceedings of the 25th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS 2006)*, Chicago, IL, June 2006, 163–172.
- (215) Richard Lau, Stephanie Demers, Yibei Ling, Bruce Siegel, Einar Vollset, Ken Birman, Robbert van Renesse, Howie Shrobe, Jonathan Bachrach, and Lester Foster. Cognitive Adaptive Radio Teams. *Proceedings of the 2006 International Workshop on Wireless Ad hoc and Sensor Networks (IWWAN 2006)*, New York, NY, June 2006.

- (216) J. Leskovec, D. Chakrabarti, J. Kleinberg, and C. Faloutsos. Realistic, Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication. *Proceedings of the European Conference on Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, 2005.
- (217) J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over Time: Den-sification Laws, Shrinking Diameters and Possible Explanations. *Proceedings 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005, 177–187.
- (218) J. Leskovec, A. Singh, and J. Kleinberg. Patterns of Influence in a Recommendation Network. *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, 2006.
- (219) David Liben-Nowell and Jon M. Kleinberg. Structural properties and tractability results for linear synteny. *J. Discrete Algorithms*, 2(2004), 207–228.
- (220) D. Liben-Nowell and J. Kleinberg. The Link Prediction Problem for Social Networks. *Proc. 12th International Conference on Information and Knowledge Management (CIKM)* (2003), 556–559.
- (221) Prakash Linga, Adina Crainiceanu, Johannes Gehrke and Jayavel Shan-mugasundaram. Guaranteeing Correctness and Availability in P2P Range Indices. *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data (SIGMOD 2005)*, June 2004, 323–334.
- (222) Prakash Linga, Indranil Gupta, and Ken Birman. A Churn-Resistant Peer-to-Peer Web Caching System. *Proceedings ACM Workshop on Survivable and Self-Regenerative Systems* (October 2003).
- (223) Hongzhou Liu, Tom Roeder, Kevin Walsh, Rimón Barr, and Emin Gün Sirer. Design and Implementation of a Single System Image Operating System for Ad Hoc Networks. *Proceedings of The International Conference on Mobile Systems, Applications and Services (MobiSys)*, Seattle, WA, June 2005, 149–162.
- (224) Hongzhou Liu and Emin Gün Sirer. A Measurement Study of RSS, A Publish-Subscribe System for Web Micronews. *Proceedings of the Internet Measurement Conference*, Berkeley, California, October 2005, 29–34.

- (225) Jed Liu, Aaron Kimball, and Andrew C. Myers. Interruptible iterators. *Proceedings of the 33th ACM Symposium on Principles of Programming Languages (POPL)*, Charleston, SC, January 2006, 283–294.
- (226) Jed Liu and Andrew C. Myers. JMatch: Iterable abstract pattern matching for Java. *Proceedings of the 5th International Symposium on Practical Aspects of Declarative Languages* (New Orleans, LA, January 2003), 110–127.
- (227) Lori Lorigo, Jon M. Kleinberg, Richard Eaton, and Robert L. Constable. A Graph-Based Approach Towards Discerning Inherent Structures in a Digital Library of Formal Mathematics. *Mathematical Knowledge Management, Third International Conference*, 2004, 220–235.
- (228) Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. l-Diversity: Privacy Beyond k-Anonymity. *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE 2006)*, Atlanta, Georgia, April 2006, 24.
- (229) Tobias Mayr, Philippe Bonnet, J. E. Gehrke, and Praveen Seshadri. Leveraging non-uniform resources for parallel query processing. *Proceedings of the Third IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2003)*, (Tokyo, Japan, May 2003).
- (230) L. Meyerguz, D. Kempe, J. Kleinberg, and R. Elber. The Evolutionary Capacity of Protein Structures. *Proceedings ACM RECOMB International Conference on Computational Molecular Biology* (2004), 290–297.
- (231) Yaron Minsky and Fred B. Schneider. Tolerating malicious gossip. *Distributed Computing* 16, 1 (Feb 2003), 49–68.
- (232) G. Morrisett. Type checking systems code. *Eleventh European Symposium on Programming* (Grenoble, France, April 2002), Lecture Notes in Computer Science Volume 2305, 1–5.
- (233) S. Muthukrishnan, Rajmohan Rajaraman, Anthony Shaheen, and Johannes Gehrke. Online Scheduling to Minimize Average Stretch. *SIAM Journal on Computing*, Vol. 34, No. 2, 2004, 433–452.
- (234) Andrew C. Myers, Andrei Sabelfeld, and Steve Zdancewic. Enforcing robust declassification and qualified robustness. *Journal of Computer Security* Vol.14, No.2, 2006, 157–196.

- (235) Andrew C. Myers, Andrei Sabelfeld, and Steve Zdancewic. Enforcing Robust Declassification. *Proceedings 17th IEEE Computer Security Foundations Workshop*, June 2004.
- (236) Alexandru Niculescu-Mizil and Rich Caruana. Obtaining Calibrated Probabilities from Boosting. *International Conference on Uncertainty in Artificial Intelligence*, August 2005, 413–420.
- (237) Alexandru Niculescu-Mizil and Rich Caruana. Predicting Good Probabilities with Supervised Learning. *International Conference on Machine Learning*, August 2005, 625–632.
- (238) Nathaniel Nystrom, Stephen Chong and Andrew C. Myers. Scalable Extensibility via Nested Inheritance. *Proceedings of the 19th ACM Conference on Object-Oriented Programming Systems, Languages and Applications*, October 2004, 99–115.
- (239) Nathaniel Nystrom, Michael R. Clarkson, and Andrew C. Myers. Polyglot: An extensible compiler framework for Java. *Proceedings of the 12th International Conference on Compiler Construction* (Warsaw, Poland, April 2003), Lecture Notes in Computer Science, Volume 2622, 138–152.
- (240) Panagiotis Papadimitratos, Zygmunt Haas, and Emin Gün Sirer. Path-Set selection in mobile ad hoc networks. *Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing* (MOBIHOC), (Lausanne, Switzerland, June 2002).
- (241) Ryan Peterson, Venugopalan Ramasubramanian, and Emin Gün Sirer. A Practical Approach to Peer-to-Peer Publish-Subscribe. *login*: 31:4, July 2006, 42–46.
- (242) S. Pleisch, M. Balakrishnan, K. Birman, and R. van Renesse. Mistral: Efficient Flooding in Mobile Ad hoc Networks. *Seventh ACM International Symposium on Mobile Ad Hoc Networking and Computing* (ACM MobiHoc 2006), Florence, Italy, May 2006.
- (243) R. Pucella. On equivalences for a class of timed regular expressions. *Proceedings of the 7th International Workshop on Coalgebraic Methods in Computer Science (CMCS'04)*, 2004.
- (244) R. Pucella. Deductive algorithmic knowledge. *Proceedings of the Eighth International Symposium on Artificial Intelligence and Mathematics (SAIM'04)*, AI&M 22-2004, 2004.

- (245) R. Pucella and V. Weissman. A formal foundation for ODRL. *Proceedings of the Workshop on Issues in the Theory of Security (WITS'04)*, 2004.
- (246) R. Pucella and V. Weissman. Reasoning about dynamic policies. *Proceedings of the 7th International Conference on Foundations of Software Science and Computation Structures (FOSSACS'04)*, Lecture Notes in Computer Science 2987, Springer-Verlag, 2004, 453–467.
- (247) R. Pucella and V. Weissman. A logic for reasoning about digital rights. *Proceedings of the 15th IEEE Computer Security Foundations Workshop* (2002), 282–294.
- (248) Jian Qiu, Feng Shao, Misha Zatsman, and Jayavel Shanmugasundaram. Index structures for querying the deep web. *Proceedings of the Workshop on Web and Data Bases* (Jun 2003), 79–86.
- (249) Venugopalan Ramasubramanian, Zygmunt J. Haas, and Emin Gün Sirer. SHARP: A hybrid adaptive routing protocol for mobile ad hoc networks. *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)* (Annapolis, Maryland, June 2003).
- (250) Venugopalan Ramasubramanian and Daniel Mosse. Statistical analysis of connectivity in unidirectional mobile ad hoc networks. *Proceedings International Workshop on Ad Hoc Networking 2002* (Vancouver, Canada), August 2002.
- (251) Venugopalan Ramasubramanian, Ryan Peterson, and Emin Gün Sirer. Corona: A High Performance Publish-Subscribe System for the World Wide Web. *Proceedings of the 3rd Symposium on Networked Systems Design and Implementation*, San Jose, California, May 2006, 15–28.
- (252) Venugopalan Ramasubramanian and Emin Gün Sirer. Perils of Transitive Trust in the Domain Name System. *Proceedings of Internet Measurement Conference*, Berkeley, October 2005, 379–384.  
  
Venugopalan Ramasubramanian and Emin Gün Sirer. The Design and Implementation of a Next Generation Name Service for the Internet. *Proceedings of the SIGCOMM Conference* (Portland, Oregon, August 2004).
- (253) Radu Rugina and Martin Rinard. Symbolic Bounds Analysis of Pointers, Array Indices, and Accessed Memory Regions. *ACM Transactions*

on *Programming Languages and Systems*, Vol. 27, No. 2, March 2005, 185–235.

- (254) Andrei Sabelfeld and Andrew C. Myers. Language-based information-flow security. *IEEE Journal on Selected Areas in Communications*, 21, No. 1 (January 2003), 5–19.
- (255) Fred B. Schneider. Lifting reference monitors from the kernel. *Formal Aspects of Security, FASec 2002* (London, United Kingdom, December 2002), Ali E. Abdullah, Peter Ryan, and Steve Schneider (eds.). Lecture Notes in Computer Science, Volume 2629, Springer-Verlag, New York, 2003, 1–2.
- (256) Fred B. Schneider. Least privilege and more. *Computer Systems: Papers for Roger Needham*, Andrew Herbert and Karen Sparck Jones, eds. Microsoft Research, 2003, 209–213.
- (257) Fred B. Schneider. Least Privilege and More. *IEEE Security and Privacy*, Volume 1, Number 3 (September/October 2003), 55–59. Also appears in *Computer Systems: Theory, Technology, and Applications*, (A. Herbert and K. Jones, eds). Springer-Verlag, New York, 253–258.
- (258) Fred B. Schneider. The Next Digital Divide. Editorial. *IEEE Security and Privacy* 2, 1 (January/February 2004), 5.
- (259) Fred B. Schneider. Time out for station identification. Editorial. *IEEE Security and Privacy* 2, 1 (September/October 2004), 5.
- (260) Fred B. Schneider. It depends on what you pay. Editorial. *IEEE Security and Privacy* 3, 3 (May/June 2005), 5.
- (261) Fred B. Schneider. Here Be Dragons. Editorial. *IEEE Security and Privacy* 4:3, May/June 2006, 3.
- (262) Fred B. Schneider and Lidong Zhou. Implementing Trustworthy Services Using Replicated State Machines. *IEEE Security and Privacy*, 3:5, September 2005, 34–43.
- (263) Jayavel Shanmugasundaram, I. Tatarinov, E. Viglas, K. Beyer, E. Shekita, and C. Zhang. Storing and querying ordered XML using a relational database system. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, (Madison, Wisconsin), May 2002.

- (264) Jayavel Shanmugasundaram, Fan Yang, Mirek Riedewald, Johannes Gehrke, and Alan Demers. Hilda: A High-Level Language for Data-Driven Web Applications. *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE 2006)*, Atlanta, Georgia, April 2006, 32.
- (265) Feng Shao, Antal Novak, and Jayavel Shanmugasundaram. Triggers over XML Views of Relational Data. *Proceedings of the IEEE International Conference on Data Engineering*, Tokyo, Japan, April 2005, 483–484.
- (266) Benyah Shaparenko, Rich Caruana, Johannes Gehrke, and Thorsten Joachims. Identifying Temporal Patterns and Key Players in Document Collections. *Proceedings of the Workshop on Temporal Data Mining: Algorithms, Theory and Applications (ICDM'05)*, Houston, Texas, November 2005, 165–174.
- (267) Alan Shieh, Andrew C. Myers, and Emin Gün Sirer. Trickle: A Stateless Network Stack for Improved Scalability, Resilience and Flexibility. *Proceedings 2nd USENIX Symposium on Networked Systems Design and Implementation*, May 2005, 175–188.
- (268) M. Singh, P. Pradhan, and P. Francis. MPAT: Aggregate TCP Congestion Management as a Building Block for Internet QoS. *Proceedings IEEE International Conference on Network Protocols (ICNP)* (Berlin, Germany, October 2004).
- (269) Emin Gün Sirer. Heuristics Considered Harmful: Using Mathematical Optimization for Resource Management in Distributed Systems. *IEEE Intelligent Systems, Special Issue on Self-Management through Self-Organization in Information Systems* March/April 2006, 55–57.
- (270) Emin Gün Sirer, Sharad Goel, Mark Robson, and Dogan Engin. Eluding Carnivores: File Sharing with Strong Anonymity. *Proceedings of European SIGOPS Workshop*, Leuven, Belgium, September 2004, 103–108.
- (271) Emin Gün Sirer and Ke Wang. A temporal logic-based access control mechanism for web services. *Proceedings of the Eighth Symposium on Access Control Models and Technologies (SACMAT)*, (Monterey, California), May 2002.

- (272) Emin Gün Sirer and Ke Wang. An access control language for web services. *Proceedings of the Symposium on Access Control Models and Technologies (SACMAT)*, (Monterey, CA, June 2002), 23–32.
- (273) Frederick Smith, Dan Grossman, Greg Morrisett, Luke Hornof, and Trevor Jim. Compiling for template-based run-time code generation. *Journal of Functional Programming*, 13(3):677–708, May 2003.
- (274) Matt Stillerman and Dexter Kozen. Demonstration: Efficient code certification for open firmware. *Proceedings 3rd DARPA Information Survivability Conference and Exposition (DISCEX III)*, volume 2, IEEE Computer Society, Los Alamitos, CA, April 2003, 147–148.
- (275) Scott D. Stoller and Fred B. Schneider. Automated analysis of fault-tolerance in distributed systems. *Formal Methods in System Design* 28, 2 (March 2005), 183–196.
- (276) Niki Trigoni, Yong Yao, Alan J. Demers, Johannes Gehrke, and Rajmohan Rajaraman. Hybrid Push-Pull Query Processing for Sensor Networks. *GI Jahrestagung*, Vol. 1, 2004, 370–374.
- (277) Niki Trigoni, Yong Yao, Alan J. Demers, Johannes Gehrke, and Rajmohan Rajaraman. Multi-query Optimization for Sensor Networks. *First IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS 2005)*, 307–321.
- (278) Robbert van Renesse. Power-Aware Epidemics. *Proceedings of the International Workshop on Reliable Peer-to-Peer Systems* (Osaka, Japan, Oct. 2002).
- (279) Robbert van Renesse. Efficient Reliable Internet Storage. *Workshop on Dependable Data Management*, Florianopolis, Brazil, October 2004.
- (280) Robbert van Renesse and Kenneth P. Birman. Autonomic Computing—A System-Wide Perspective. *Autonomic Computing: Concepts, Infrastructure, and Applications*, ed. Manish Parashar and Salim Hariri, CRC press, 2006, 35–48.
- (281) Robbert van Renesse, Ken Birman, Adrian Bozdog, Dan Dumitriu, Manpreet Singh and Werner Vogels. Heterogeneity-Aware Peer-to-Peer Multicast. *Proceedings of the 17th International Symposium on Distributed Computing (DISC 2003)*. (Sorrento, Italy, October 2003).

- (282) Robbert van Renesse, Kenneth Birman, Dan Dumitriu, and Werner Vogels. Scalable management and data mining using Astrolabe. *Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS)*, (Cambridge, Massachusetts), March 2002.
- (283) Robbert van Renesse, Kenneth Birman, and Werner Vogels. Astrolabe: A Robust and Scalable Technology for Distributed System Monitoring, Management, and Data Mining. *ACM Transactions on Computer Systems* 21, 2 (May 2003) 164–206.
- (284) Robbert van Renesse and Fred B. Schneider. Chain Replication for Supporting High Throughput and Availability. *Sixth Symposium on Operating Systems Design and Implementation*, (OSDI 04), San Francisco, CA, December 2004, 91–104.
- (285) Vidhyashankar Venkataraman, Paul Francis, and John Calandrino. Chunkyspread: Multi-tree Unstructured Peer to Peer Multicast. *Proceedings of the 5th International Workshop on Peer-to-Peer Systems*, Santa Barbara, California, February 2006.
- (286) Muthuramakrishnan Venkitasubramaniam, Ashwin Machanavajjhala, David Martin, and Johannes Gehrke. Trusted CVS. *Proceedings of the International Workshop on Security and Trust in Decentralized/Distributed Data Structures (STD3S)*, Atlanta, GA, April 2006, 23.
- (287) Vivek Vishnumurthy, Sangeeth Chandrakumar, and Emin Gün Sirer. KARMA: A secure economic framework for peer-to-peer resource sharing. *Proceedings of the Workshop on Economics of Peer-to-Peer Systems* (Berkeley, California, June 2003).
- (288) Vivek Vishnumurthy and Paul Francis. On Overlay Construction and Random Node Selection in Heterogeneous Unstructured P2P Networks. *Proceedings of IEEE Infocom 2006*, Barcelona, April 2006.
- (289) Werner Vogels. Architectural Challenges for Global Information Systems. *Proceedings of the High Performance Transaction Systems Workshop* (Asilomar, CA, October 2003).
- (290) Werner Vogels. Architectural challenges for global information systems. *Proceedings of the 15th Supercomputing Conference (SC2003)* (Phoenix, Arizona, November 2003).

- (291) Werner Vogels. HPC.NET—Are CLI based Virtual Machines Suitable for High Performance Computing? *Proceedings of the 15th Supercomputing Conference (SC2003)* (Phoenix, Arizona, November 2003).
- (292) Werner Vogels and Chris Re. WS-Membership – Failure management in a web-services world. *Proceedings 12th International World Wide Web Conference* (Budapest, Hungary, May 2003).
- (293) Werner Vogels, Chris Re, Robbert van Renesse, and Ken Birman. A collaborative infrastructure for scalable and robust news delivery. *Proceedings of the IEEE Workshop on Resource Sharing in Massively Distributed Systems (RESH'02)*, (Vienna, Austria), July 2002.
- (294) Werner Vogels, Robbert van Renesse and Ken Birman. The power of epidemics: Robust communication for large-scale distributed systems. *Proceedings of HotNets-I '02: First Workshop on Hot Topics in Networks* (Princeton, NJ, Oct 2002).
- (295) Kevin Walsh, Emin Gün Sirer. Staged Simulation for Improving the Scale and Performance of Wireless Network Simulations. *Proceedings of the Winter Simulation Conference* (New Orleans, LA, December 2003).
- (296) Kevin Walsh and Emin Gün Sirer. Staged Simulation: A General Technique for Improving Simulation Scale and Performance. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, Special Issue on Scalable Network Modeling and Simulation, April 2004.
- (297) Kevin Walsh and Emin Gün Sirer. Fighting Peer-to-Peer SPAM and Decoys with Object Reputation. *Proceedings of the Workshop on the Economics of Peer-to-Peer Systems (P2PECON)*, Philadelphia, PA, August 2005, 138–143.
- (298) Kevin Walsh and Emin Gün Sirer. Experience with an Object Reputation System for Peer-to-Peer Filesharing. *Proceedings of 3rd Symposium on Networked Systems Design and Implementation*, San Jose, California, May 2006, 1–14.
- (299) Stephanie Weirich. Higher-order intensional type analysis. *Eleventh European Symposium on Programming* (Grenoble, France, April 2002), Lecture Notes in Computer Science Volume 2305, 98–114.
- (300) Stephanie Weirich. *Programming With Types*. Ph.D. Thesis, Cornell University, Department of Computer Science, July 2002.

- (301) V. Weissman and C. Lagoze. Towards a Policy Language for Humans and Computers. *ECDL-04 (Proceedings of the Eighth European Conference on Digital Libraries)*, 2004, 513–525.
- (302) Dan Williams and Emin Gün Sirer. Optimal Parameter Selection for Efficient Memory Integrity Verification Using Merkle Hash Trees. *Proceedings of the Trustworthy Network Computing Workshop*, August 2004.
- (303) Bernard Wong and Emin Gün Sirer. ClosestNode.com: An Open-Access, Scalable, Shared Geocast Service for Distributed Systems. *SIGOPS Operating Systems Review* 40:1, January 2006, 62–64.
- (304) Bernard Wong, Aleksandrs Slivkins, and Emin Gün Sirer. Meridian: A Lightweight Network Location Service Without Virtual Coordinates. *Proceedings of SIGCOMM Conference*, Philadelphia, PA, August 2005, 85–96.
- (305) Bernard Wong, Ivan Stoyanov, and Emin Gün Sirer. Geolocalization on the Internet through Constraint Satisfaction. *Workshop on Real, Large Distributed Systems*, Seattle, November 2006.
- (306) Yong Yao and J. E. Gehrke. The Cougar approach to in-network query processing in sensor networks. *SIGMOD Record*, Vol 31, No 3, September 2002.
- (307) Yong Yao and J. E. Gehrke. Query processing in sensor networks. *Proceedings of the First Biennial Conference on Innovative Data Systems Research (CIDR 2003)* (Asilomar, California, January 2003).
- (308) I. V. Yap, D. Schneider, J. Kleinberg, D. Matthews, S. Cartinhour, and S. R. McCouch. A Graph-Theoretic Approach to Comparing and Integrating Genetic, Physical and Sequence-Based Maps. *Genetics*, 165 (2003).
- (309) Steve Zdancewic and Andrew C. Myers. Secure information flow and linear continuations. *Higher Order and Symbolic Computation* 15, Nos. 2–3 (Sept. 2002), 209–234.
- (310) Steve Zdancewic and Andrew C. Myers. Observational determinism for concurrent program security. *Proceedings of the 16th IEEE Computer Security Foundations Workshop* (Pacific Grove, California, June 2003), 29–43.

- (311) Lantian Zheng, Stephen Chong, Andrew C. Myers, and Steve Zdancewic. Using replication and partitioning to build secure distributed systems. *Proceedings of the IEEE Symposium on Security and Privacy* (Oakland, California, May 2003), 236–250.
- (312) Lantian Zheng and Andrew C. Myers. Dynamic Security Labels and Noninterference. *Proc. 2nd Workshop on Formal Aspects in Security and Trust*, August 2004.
- (313) Lantian Zheng and Andrew Myers. End-to-end Availability Policies and Noninterference. *Proceedings 18th IEEE Computer Security Foundations Workshop*, June 2005, 272–286.
- (314) Lidong Zhou and Fred B. Schneider. APSS: Proactive secret sharing in asynchronous systems. *ACM Transactions on Information and System Security* 8, 3 (August 2005), 1–28.